Model 102-A
GENERAL PURPOSE COMPUTER

OPERATION & MAINTENANCE MANUAL VOLUME II

The National Cash Register Company
Dayton, Ohio

model
102-A
GENERAL
PURPOSE
COMPUTER

operation and maintenance manual volume II

THE NATIONAL CASH REGISTER COMPANY
DAYTON, OHIO

Copyright 1956
©The National Cash Register Company
Dayton, Ohio

FOREWORD

This instruction manual is provided as a basic source of technical information on the Model 102-A General Purpose Computer manufactured by The National Cash Register Company, Dayton, Ohio. This manual also includes a description of the supplementary input-output equipment which can be used in this computing system. For convenience, the manual is contained in two separate volumes.

Volume I contains descriptive data, installation procedures, explanations of the theory of operation, and operating maintenance instructions. Volume II contains the schematics, logical networks and equations, and the block explanations. The information provided in these two volumes is sufficiently detailed to enable qualified personnel to operate and completely service the computer.

Supplementary input-output equipment obtained from vendors other than this company, are modified, as required, for operation in this computing system. These modifications are made by this company and are described in detail along with the operation and maintenance of this equipment as part of the 102-A system. Appropriate publications are referenced for more complete information on these units.

Table of Contents

FOREWORD V

BLOCK EXPLANATIONS 1

SCHEMATICS

Figure 1	Flow Diagram	211
Figure 2	A_1 Logical Networks	213
Figure 3	A_2 Logical Networks	215
Figure 4	A_3 and A_4 Logical Networks	217
Figure 5	A_5 and A_6 Logical Networks	219
Figure 6	A_7 through A_{12} Logical Networks	221
Figure 7	Pulse and Digit Counter Logical Networks	223
Figure 8	k Logical Networks	225
Figure 9	$_0$k Logical Networks	227
Figure 10 (1 of 2)	Program Counter Grid Networks	229
Figure 10 (2 of 2)	Program Counter Grid Networks	231
Figure 11 (1 of 3)	Program Counter Matrix Networks	233
Figure 11 (2 of 3)	Program Counter Matrix Networks	235
Figure 11 (3 of 3)	Program Counter Matrix Networks	237
Figure 12 (1 of 3)	Program Counter Sum Networks	239
Figure 12 (2 of 3)	Program Counter Sum Networks	241
Figure 12 (3 of 3)	Program Counter Sum Networks	243
Figure 13	L_1 through L_5 Channel Selector Logical Networks	245
Figure 14	Memory Logical Networks	247
Figure 15	Recirculating Registers	249
Figure 16	E_o Logical Network	251
Figure 17	F_o and G_o Logical Networks	253
Figure 18	H_o Logical Networks	255
Figure 19 (1 of 2)	J_o Logical Network	257
Figure 19 (2 of 2)	J_o Logical Network	259
Figure 20 (1 of 3)	Power Supply Schematic	261
Figure 20 (2 of 3)	Power Supply Schematic	263
Figure 20 (3 of 3)	Power Supply Schematic	265
Figure 21 (1 of 3)	Plug-in Schematics	267
Figure 21 (2 of 3)	Plug-in Schematics	269
Figure 21 (3 of 3)	Plug-in Schematics	271
Figure 22	Test Panel Schematic	273
Figure 23	Memory Tube Chassis Component Installation	275
Figure 24	Physical Locations of Read-Write Heads	277
Figure 25 (1 of 2)	Schematic of Operator's Console	279
Figure 25 (2 of 2)	Schematic of Operator's Console	281
Figure 26	Input-Output Circuitry	283
Figure 27	Flexowriter Conversion Circuits	285
Figure 28 (1 of 2)	Thyratron Grid Logical Networks	287
Figure 28 (2 of 2)	Thyratron Grid Logical Networks	289

Model 102-A System

Count in from PC # ----

Skip in from PC # 35, 40, 231, 430. with K true

Explanation:

Clear switch sets PC to this block and resets A_1-A_7.

Flexowriter signal or start button (S) on control panel and sampling pulse (SS) set A_1-A_5. And S F. F. which sets K false for counting out.

E, F, G, H, J recirculate.

All numbers shown in a_1-a_5 are FX (Flexowriter numbers, not PC numbers. The Flexowriter code numbers represent the 6 holes across the tape as two (2) octal digits, as indicated below.

Character	Flexowriter Code	Computer Code A_5	A_4	A_3	A_2	A_1
0	37	0	0	0	0	0
1	11	0	0	0	0	1
2	74	0	0	0	1	0
3	70	0	0	0	1	1
4	64	0	0	1	0	0
5	62	0	0	1	0	1
6	66	0	0	1	1	0
7	72	0	0	1	1	1
8	60	0	1	0	0	0
9	33	0	1	0	0	1
+	52	0	0	0	0	0
Minus	56	0	0	0	1	0
Period	42	0	1	1	1	1
Space	04	0	1	1	0	0
f	26	1	0	0	0	0
d	22	1	0	0	0	1
o (letter)	03	1	1	0	0	0
Hyphen	50	1	1	0	0	0
Tab	51	1	1	1	0	0
s	24	1	1	1	1	1

Count out to PC # 1 (Character or Control) function of blk #1

Skip out to PC # Stick in 0 (Rest)

BLOCK / WORD-TIME / PROPOSITION CHART

0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B-2
F_0	←						F_2							→	7D-3
G_0	←						G_5							→	7B-2
H_0	←						H_2							→	7A-1
J_0	←						J_2							→	8A-5
$_0k$	←						KS							→	9A-4
a_1		[FX	(11	+70	+62	+72	+33	+42	+22	+24)	+	Ⓢ]	(SS)		12D-7
a_2		[FX	(74	+70	+66	+72	+56	+42	+51	+29)	+	Ⓢ]	(SS)		12C-7
a_3		[FX	(64	+62	+66	+72	+04	+50	+51	+24	+42)	+ Ⓢ]	(SS)		12B-8
a_4		[FX	(60	+33	+04	+42	+03	+50	+51	+14	+22)	+ Ⓢ]	(SS)		12A-7
a_5			[FX	(22	+03	+24	+50	+51	+24)	+	Ⓢ]	(SS)			18D-1

BLOCK FUNCTION: Rest

MAIN ROUTINE ---

0

PC BLOCK #

Count in from PC # 0 (Rest)

Skip in from PC # ----

Explanation:

E, H, J recirculate. F.G. clearing

Test (A_5) for control signal, or character to be filled.

Control signal: (A_5) = 1, K is left false, count to PC 2.

Character: (A_5) = 0, K is set true, skip to PC 41.

Count out to PC # 2 Test (A_4)

Skip out to PC # 41 Fill A_1-A_4 into E

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←						E_5	→							6B-2
H_o	←						H_2	→							7A-1
J_o	←						J_2	→							8A-5
k	←						A'_5	→							9B-4

BLOCK FUNCTION: Test on (A) for Control Signal or Character

MAIN ROUTINE Fill

1

PC BLOCK #

Count in from PC # 1 Test on A_5 for Control or Character

Skip in from PC # ---

Explanation:

E, H, J recirculate.

Test (A_4) for shifting (E) left four digits.

If (A_4) is true or (A_4) = 1, K is left false, count to PC 3.

If A_4 = 0, K is set true, skip to PC 54 and shift (E) left four digits.

Count out to PC # 3 Decimal or Octal, Test on (A_3).

Skip out to PC # 54 Shift (E) left

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	←	←	←	←	←	←	E_5	→	→	→	→	→	→	6B-2
H_0	←	←	←	←	←	←	←	H_2	→	→	→	→	→	→	7A-1
J_0	←	←	←	←	←	←	←	J_2	→	→	→	→	→	→	8A-5
$\not{k}$	←	←	←	←	←	←	←	A_4'	→	→	→	→	→	→	9B-4

BLOCK FUNCTION: Test on (A_4) for shifting (E)

MAIN ROUTINE Fill

2

PC BLOCK #

Count in from PC # 2 Test (A_4)

Skip in from PC # ----

Explanation:

E, H, J recirculate.

Test (A_3) for number base change (decimal or octal).

If (A_3) = 1, K is left false, count to PC 4.

(A_3) = 0, K is set true, skip to PC 43.

Count out to PC # 4 Test (A_2)

Skip out to PC # 43 Set A_6 if Decimal

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←	—	—	—	—	—	—	E_5	—	—	—	—	—	→	6B-2
H_o	←	—	—	—	—	—	—	H_2	—	—	—	—	—	→	7A-1
J_o	←	—	—	—	—	—	—	J_2	—	—	—	—	—	→	8A-5
k	←	—	—	—	—	—	—	A_3'	—	—	—	—	—	→	9B-4

BLOCK FUNCTION: Test on (A_3) for number base change

MAIN ROUTINE Fill

3

PC BLOCK #

Count in from PC # 3 Test (A_3)

Skip in from PC # ---

Explanation:

E, H, J recirculate.

Test (A_2) for hyphen.

If (A_2) = 1, K is left false, count to PC 5.

If (A_2) = 0, K is set true, skip to PC 44.

Count out to PC # 5 Test (A_1)

Skip out to PC # 44 Fill (E) into H

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	—	E_5	—	—	—	—	—	→	6B-2
H_0	←	—	—	—	—	—	—	H_2	—	—	—	—	—	→	7A-1
J_0	←	—	—	—	—	—	—	J_2	—	—	—	—	—	→	8A-5
k	←	—	—	—	—	—	—	A'_2	—	—	—	—	—	→	9B-4

BLOCK FUNCTION: Test (A_2) for hyphen

MAIN ROUTINE Fill

4

PC BLOCK #

Count in from PC # 4 Test (A_2)

Skip in from PC # ----

Explanation:

E, H, J recirculate.

Test (A_1) for Tab.

If (A_1) = 1, K is left false and count to PC 6.

If (A_1) = 0, K is set true and skip to PC 45.

Count out to PC # 6 Select channel in m_2 of H

Skip out to PC # 45 Select channel in m_3 of H

set N true 100101

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←							E_5						→	6B-2
H_0	←							H_2						→	7A-1
J_0	←							J_2						→	8A-5
k	←							A_1'						→	9B-3

BLOCK FUNCTION: Test (A_1) for Tab

MAIN ROUTINE Fill

5

PC BLOCK #

Count in from PC # 5 Test (A_1) for Tab

Skip in from PC # 66, 67, 136, 137, 233, 407, 447, 467, 517, 537, 566, 624, 637

Explanation:

H and J recirculate.

Set flip-flops L_1-L_5 to select channel according to the control number in 0_{6-7} of m_2 of H. L_1 is set to retain (P_0 0_6) of H; L_2 to retain (P_1 0_6) of H; etc.

K is set false for count to PC 7.

Count out to PC # 7 Select Sector in m_2 of H

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	
H_0		←	—	—	—	—	—	—	H_2	—	—	—	—	—	→	7A-1
J_0		←	—	—	—	—	—	—	J_2	—	—	—	—	—	→	8A-5
$_0k$													///			9A-4
l_1									H_2							10D-1,2
$_0l_1$									H_2'							10D-1,2
l_2									H_2							10D-1, C-2
$_0l_2$									H_2'							10D-1, C-2
l_3									H_2							10D-1, C-2
$_0l_3$									H_2'							10D-1, B-2
l_4								H_2								10D-1, B-2
$_0l_4$								H_2'								B-2
l_5								H_2								A-2
$_0l_5$								H_2'								A-2

BLOCK FUNCTION: Select channel in m_2 of H

MAIN ROUTINE Locating Command

6

PC BLOCK #

Count in from PC # 6 (Select channel in m_2 of H)

Skip in from PC # ----

Explanation:

H and J recirculate.

K is set false by X', which is true except in the special testing configuration, and K is set true if digits in 0_{4-5} of M_w do not coincide with 0_{4-5} of H. Only 0_4 of H is tested if look-up is in buffer (J_2), that is, when $L_5L_4'L_1'$ is true in equation 2.

K cannot be set true for 3000 or 2100 address (L_5 = 1) (L_4 or L_1 = 1).

When M_w and H digits coincide, K is left false resulting in count to PC 10.

$M_w \cdot H_2$ different turn out

$M_w \cdot H_2$ came, OK to count out

Count out to PC # 10 Read (m_2) into H and t.o.

Skip out to PC # Stick in PC 7 until sector is found.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
H_0	←	—	—	—	—	—	—	H_2	—	—	—	—	—	→	7A-1
J_0	←	—	—	—	—	—	—	J_2	—	—	—	—	—	→	8A-5
k									①	②					9D-4
$_0k$	X'														9A-4
				①	$k = (M_w$	$H_2' +$	M_w'	$H_2)$	L_5'	0_{4-5}					Main memory
				②	$k = (M_w$	$H_2' +$	M_w'	$H_2)$	L_5	$L_4' L_1' 0_4$					2000

BLOCK FUNCTION: Select Sector of m_2 of H

MAIN ROUTINE Locating Command

7

PC BLOCK #

Count in from PC # 7 Select sector in m_2 of H

Skip in from PC # ----

Explanation:

J recirculates.

Read command into H from M if (L_1-L_5) is 00-17; from J_2 (buffer) if 20; from G is 30.

Add 1 to control number and place in m_2 (0_{4-7}) of G.

K is set true in 0_2 for adding 1 to (m_2) of H, the control number.

If m_2 of H happens to read 7777 so that m_2 of H'_2 is all zeros, K will remain true, causing skip to PC 434. Any other m_2 will set K false before 0_{12}.

Testing for overflow. The overflow test sets K true to skip to PC 434 (error print) if

(1) The preceding order has resulted in an overflow (A_7) = 1 and

(2) The instruction is not S1-27) or to (37) (i.e. a zero in 0_{12} or $P_1 0_{13}$) and

(3) The automatic overflow switch is on.

$H_0 \leftarrow M\ L_5'$ (memory)

$\leftarrow J_2\ L_5 L_4' L_1'$ (2000)

$\leftarrow G_5 L_5 L_4$ (3000)

$\leftarrow L_5 L_4' L_1$ 2100

$\rightarrow$ K during $0_{12} + P_1 0_{13}$

A_7 of M'

Count out to PC # 11 Set up Channel selector to find m_1

Skip out to PC # 434 Set up to print control number

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
							(unit clocks)								
G_0	←		H_2				$KH'_2 + K'H_2$					H_2		→	7C-3
H_0	←		$ML'_5 + J_2$	$L_5 L'_4 L'_1$			$+ G_5 L_5 L_4$							→	7A-1
J_0	←					J_2									8A-5
K	①	①										III			9D-2
o K							←	H'_2		→					9A-5
			①	K =	$(M'L'_5 + J'_2 L_5)$	SW	$A_7 (0_{12} + P_1 0_{13})$								Error print only
			27 37						S.F. reset on						
		from M_M								from buffer					

BLOCK FUNCTION: Read command into H and t.o.

MAIN ROUTINE Locating command

10

PC BLOCK #

Count in from PC # 10 Read command into H and t.o.

Skip in from PC # ----

Explanation: E line clearing + F line

G, H and J recirculate.

A_1-A_{12} are set false in 0_0 to clear all A flip-flops.

The L flip-flops are set to select the channel from m_1 ($0_{10\text{-}11}$) of H.

K remains false.

Count out to PC # 12 Find the sector in m_1 of H

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
	G_0	←	—	—	—	—	—	—	G_5	—	—	—	—	—	→	7B-2
	H_0	←	—	—	—	—	—	—	H_2	—	—	—	—	—	→	7A-1
	J_0	←	—	—	—	—	—	—	J_2	—	—	—	—	—	→	8A-5
$_0a_1$ through	$_0a_4$														///	12D1, C3, B2, A3
$_0a_5$ through	$_0a_{12}$														///	18 C3, B3, A3, A4, A2
	l_1				H_2											D2
	$_0l_1$				H_2'											D2
	l_2			H_2	H_2											D2, C2
	$_0l_2$			H_2'	H_2'											D2, C2
	l_3				H_2											10 D2, C2
	$_0l_3$				H_2'											D2, B2
	l_4			H_2												D2, B2
	$_0l_4$			H_2'												D2, B2
	l_5			H_2												D2, A2
	$_0l_5$			H_2'												D2, A2

BLOCK FUNCTION: Find Channel of m_1 of H

MAIN ROUTINE Put (m_1) into E

11

PC BLOCK #

Count in from PC # 11 Find channel m_1 of H

Skip in from PC # ----

Explanation:

G, H and J recirculate.

K enters PC 12 false and goes true for sticking if digits in 0_{8-9} of M_w and H do not coincide. Only 0_8 of H is tested if look-up is in J (buffer) as shown by $L_5L_4'L_1'$ of equation 2. K cannot be set true for 3000 or 2100 address (L_5 = 1) (L_4 or L_1 = 1).

K is reset false by X' When digits in M_w and H coincide, K is left false for count to PC 13.

M_w H_2 different time on k

Count out to PC # 13 Read (m_1) into E

Skip out to PC # Stick in PC 12 until sector is found

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
G_o	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k					←	①,②	→								9D4
$_ok$	X'														9A4
				①	$k = (M_w H_2' + M_w' H_2)$					$L_5' O_{8-9}$						Memory
				②	$k = (M_w H_2' + M_w' H_2)$					$L_5' L_4' L_1' O_8$						2000

L_5 L_4' L_1' O_8

BLOCK FUNCTION: Find Sector of m_1 of H

MAIN ROUTINE Command: Read (m_1) into E

12

PC BLOCK #

Count in from PC # 12 Find Sector of m_1 of H

Skip in from PC # ----

Explanation:

G, H and J recirculate.

Read (m_1) into E from M if (L_5) =0 (channels 0-17); from J by the proposition $L_5L'_4L'_1$; and from G if L_5 and L_4 are true (cell 3000), that is, (L_5) =1 and (L_4) =1.

Read M into E line

Count out to PC # 14 Find channel of m_2 of H

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←		ML'_5+J_2		L_5	L'_4	L'_1+G_5		L_5	L_4	→				6A5
G_o	←						G_5				→				7B2
H_o	←						H_2				→				7A1
J_o	←						J_2				→				8A5

BLOCK FUNCTION: Read (m_1) into E

MAIN ROUTINE Read (m_1) into E

13

PC BLOCK #

Count in from PC # 13 Read (m_1) into E

Skip in from PC # -----

Explanation:

E, G, H, J recirculate.

The L flip-flops are set to select the channel from m_2 (0_{6-7}) of H.

Count out to PC # 15 Find Sector of m_2 of H

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION	
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210		
E_o		←	←	←	←	←	←	←	E_5	→	→	→	→	→	→	6B2	
G_o		←	←	←	←	←	←	←	G_5	→	→	→	→	→	→	7B2	
H_o		←	←	←	←	←	←	←	H_2	→	→	→	→	→	→	7A1	
J_o		←	←	←	←	←	←	←	J_2	→	→	→	→	→	→	8A5	
l_1									H_2								D1, 2
$_o l_1$									H_2'								D1, 2
l_2									H_2								D1, C2
$_o l_2$									H_2'								D1, C2
l_3									H_2							10	D1, C2
$_o l_3$									H_2'								D1, B2
l_4								H_2									D1, B2
$_o l_4$								H_2'									D1, B2
l_5								H_2									D1, A2
$_o l_5$								H_2'									D1, A2

BLOCK FUNCTION: Find channel of m_2 of H

MAIN ROUTINE Read (m_2) into F

14

PC BLOCK #

Count in from PC # 14 Find channel of m_2 of H

Skip in from PC # -----

Explanation:

E, G, H and J recirculate.

Same explanation as PC 12 except that search is in 0_{4-5}.

BLOCK / WORD-TIME / PROPOSITION CHART

O / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
G_o	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k									←	①					9D4
										②					
$_o k$	X′														9A4
				①	$k=(M_w H_2' + M_w' H_2) L_5' O_{4-5}$										
			②	$k=(M_w H_2' + M_w' H_2) L_5 L_4' L_1' O_4$											

Count out to PC # 16 Read (m_2) of H into F

Skip out to PC # Stick in PC 15 until sector of m_2 is found

BLOCK FUNCTION: Find sector of m_2 of H

MAIN ROUTINE Read (m_2) into F

15

PC BLOCK #

Count in from PC # 15 Find Sector of m_2 of H

Skip in from PC # ----

Explanation:

E, G, H and J recirculate.

Read (m_2) into F from M if L_5 is false (channels 0-17); from J (buffer) if L_5 is true and L_4 and L_1 false; from G if L_5 and L_4 are true (cell 3000).

K is set false for the next block at $P_2 0_{13}$. However, the preceding bit $P_1 0_{13}$, which is the MSBD of I (Instruction of H_2), determines count or skip. If $P_1 0_{13}$, H_2 is zero, K is set true for one pulse and PC skips to PC 400. If $P_1 0_{13}$, H_2 is 1, K remains false and counts to PC 17.

Command

3 4
011 100

2 6
010 110

1 7
001 111

less than 20 turn
one K

Count out to PC # 17 Shift Control Number into H

Skip out to PC # 400 (Test for ts)

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←							E_5	—					→	6B2
F_o	←		$ML_5' + J_2 L_5 L_4' L_1' + G_5 L_5 L_4$											→	7D4
G_o	←							G_5	—					→	7B2
H_o	←							H_2	—					→	7A1
J_o	←							J_2	—					→	8A5
k	H_2'														9C4
k	X'														9A4

BLOCK FUNCTION: Read (m_2) into F-Command split

MAIN ROUTINE Read m_2 into F

16

PC BLOCK #

Count in from PC # 16 Read (m_2) into F; Command split

Skip in from PC # ----

Explanation:

E, F, G and J recirculate.

Put (M_2) of G, the control number, back into m_2 of H.

Count out to PC # 20 Test for command to

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	—	E_5	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	—	F_2	—	—	—	—	—	→	7D4
G_0	←	—	—	—	—	—	—	G_5	↓	—	—	—	—	→	7B2
H_0	←	—	H_2	—	—	→	←	G_5	—	→	←	H_2	—	→	7A4
J_0	←	—	—	—	—	—	—	J_2	—	—	—	—	—	→	8A5

BLOCK FUNCTION: Shift Control Number into H

MAIN ROUTINE ----

17

PC BLOCK #

Count in from PC # 17 Read (M) into F

Skip in from PC # ----

Explanation:

Set K true in 0_2.

E, F, G and J recirculate. H recirculates during $0_{0\text{-}11}$.

Equations 1 and 2 cause 1 to be added to whatever number is in $0_{12} + P_0 0_{13}$ of H.

When overflow occurs, K is left true, causeing the PC to skip out of that block. In the previous blocks, K was left false.

PC No.	Command No.	K true Skips PC No.	Operation
20	37	60	Test for overflow.
21	36	121	Subtract.
22	35	122	Add.
23	34	63	Test magnitude.
24	33	64	Test algebraically.
25	32	75	Extract.
26	31	107	Scale factor.
27	30	132	Shift Magnitude.
30	27	130	Shift logically.
31	26	171	Multiply double length.
32	25	172	Multiply and round off.
33	24	113	Divide with remainder.
34	23	114	Divide and round off.
35	22	0	Halt.
36	21	236	Print.
37	20	434	Extra (Set up to print control number if K false).

Count out to PC # See Explanation Chart above

Skip out to PC # See Explanation Chart above

BLOCK / WORD-TIME / PROPOSITION CHART

O / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←					E_5								→	6B-2
F_o	←					F_2								→	7D-3
G_o	←					G_5								→	7B-2
H_o		①	←			H_2								→	7A-3
J_o	←					J_2								→	8A-5
k												///			9B-2
$_ok$		②													9A-5

① $H_o = H_2 K' + H_2' K$

② $_oK = H_2'$

BLOCK FUNCTION Test (H) for Command

MAIN ROUTINE Command Identification

20-27, 30-37

PC BLOCK #

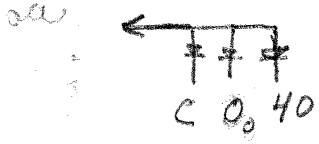
Count in from PC # -----

Skip in from PC # 41, 43, 44, 50, 57

Explanation:

E, H and J recirculate.

A_1-A_5 are set false, so that new information can be inserted by FX.



Count out to PC # -----

Skip out to PC # 0 (Rest)

BLOCK / WORD-TIME / PROPOSITION CHART

	0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B-2
H_0	←	—	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A-1
J_0	←	—	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A-5
$_0a_1$															///	D-1
$_0a_2$															///	12 C-3
$_0a_3$															///	B-2
$_0a_4$															///	A-3
$_0a_5$															///	18 C-3

BLOCK FUNCTION: Reset A_1-A_5 to 0

MAIN ROUTINE: Fill

40

PC BLOCK #

Count in from PC # ----

Skip in from PC # 1 Test on (A_5)

Explanation:

H and J recirculate.

Fill from the A flip-flops into E as follows:

(1) from A_1-A_3 if filling octally. A_6 false, A_3 follows E_5.

(2) from A_1-A_4 if filling decimally. Here A_6 is true and A_3 follows A_4, which follows E_5.

Count out to PC # ----

Skip out to PC # 40 Reset A_1-A_5

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	←	←	←	←	←	←	A_1	→	→	→	→	→	→	6A-2
H_0	←	←	←	←	←	←	←	H_2	→	→	→	→	→	→	7A-1
J_0	←	←	←	←	←	←	←	J_2	→	→	→	→	→	→	8A-5
a_1	←	←	←	←	←	←	←	A_2	→	→	→	→	→	→	D-8
$_0a_1$	←	←	←	←	←	←	←	A_2'	→	→	→	→	→	→	D-4
a_2	←	←	←	←	←	←	←	A_3	→	→	→	→	→	→	C-8
$_0a_2$	←	←	←	←	←	←	←	A_3'	→	→	→	→	→	→	C-4
a_3	←	←	←	←	←	∅	E_5	$A_6'+$	A_4	A_6	D	→	→	→	B-7
$_0a_3$	←	←	←	←	←	←	E_5'	$A_6'+$	A_4'	A_6	→	→	→	→	B-3
a_4	←	←	←	←	←	←	←	E_5	→	→	→	→	→	→	A-9
$_0a_4$	←	←	←	←	←	←	←	E_5'	→	→	→	→	→	→	A-4

12

BLOCK FUNCTION: End Octal or Decimal Digit into E.

MAIN ROUTINE: Fill

41

PC BLOCK #

Count in from PC # ----

Skip in from PC # 3 Test (A_3)

Explanation: Clear E line →

H and J recirculate.

Clear E line.

A_6 follows A_1 and is set true if decimal information is to follow, false if octal characters follow.

Count out to PC #

Skip out to PC # 40 Reset A_1-A_5

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←							0						→	
H_0	←							H_2						→	7A-1
J_0	←							J_2						→	8A-5
a_6	←							A_1						→	18B-4
$_0a_6$	←							A_1'						→	18B-4

BLOCK FUNCTION: Set A_6 if Decimal

MAIN ROUTINE: Fill

43

PC BLOCK #

Count in from PC # ----

Skip in from PC # 4 Test on (A_2)

Explanation: E -> H

J recirculates.

Hyphen character sends PC through this block and signifies that digits just entered are control addresses, not numbers to be entered into memory. (E) are transferred into H, the control register; and E is cleared.

Count out to PC # ----

Skip out to PC # 40 Clear A_1-A_5

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0															
H_0	←							E_5						→	7A-1
J_0	←							J_2						→	8A-5

BLOCK FUNCTION: Transfer (E) into H

MAIN ROUTINE: Fill

44

PC BLOCK #

M_1 M_2 M_3

Count in from PC # ----

Skip in from PC # 5 Test (A_1)

Explanation:

E, H and J recirculate.

Set K false for count to PC 46.

Set L_1-L_5, the channel selector flip-flops, as designated by m_3 of H for entering the word into the memory. The "tab" character initiates this operation.

Count out to PC # 46 Find sector of m_3 of H

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION	
E_o		←	—	—	—	—	—	—	E_5	—	—	—	—	—	→	6B-2	
H_o		←	—	—	—	—	—	—	H_2	—	—	—	—	—	→	7A-1	
J_o		←	—	—	—	—	—	—	J_2	—	—	—	—	—	→	8A-5	
$_ok$													111			9A-4	
l_1													H_2				D1-2
$_ol_1$													H_2'				D1-2
l_2													H_2				D1, C2
$_ol_2$													H_2'				D1, C2
l_3													H_2			10	D1, B2
$_ol_3$													H_2'				D1, B2
l_4												H_2					B1-2
$_ol_4$												H_2'					B1-2
l_5												H_2					B1, A2
$_ol_5$												H_2'					B1, A2

BLOCK FUNCTION: Find channel

MAIN ROUTINE: Fill

45

PC BLOCK #

Count in from PC # 45 Find channel of m_3 of H

Skip in from PC #

Explanation:

E, H and J recirculate.

The action of the computer in this block is to find the main memory sector specified by m_3 of H.

K is set true by equation ① or ② until the desired sector is found, and as a result sticks in this block until the sector is selected. When that happens, K remains false, causing the PC to count to Block 47.

At $P_1 0_{13}$ of the last word time in this block (46), the record control flip-flop, R_1, is set true if the computer is recording into any of the main memory channels, numbered 00 to 17, as determined previously in Block 45.

$P_1 O_{13}$ ⑤

$M_W \cdot H_2'$ different time on K to stick

$P_1 O_{13}$ turn off K to come back in with K off

$M_W \cdot H_2'$ same as leave K off & turn on R_1

Count out to PC # 47 Record (E) in memory

Skip out to PC # Stick in PC 46 until sector is found

BLOCK / WORD-TIME / PROPOSITION CHART

O / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←	—	—	—	—	—	—	E_5	—	—	—	—	—	→	6B-2
H_o	←	—	—	—	—	—	—	H_2	—	—	—	—	—	→	7A-1
J_o	←	—	—	—	—	—	—	J_2	—	—	—	—	—	→	8A-5
k													①	② →	9D-5
ok	X'														9A-4
r_1	③														19B-1
				①	k=	$(M_W$	H_2'+	M_W'	$H_2)$	L_5'	O_{0-1}				memory
				②	k=	$(M_W$	H_2'+	M_W'	$H_2)$	$L_5 L_4'$	O_0				3030
				③	r_1=	K'	L_5'	$P_1 O_{13}$							

BLOCK FUNCTION: Find receiver Cell

MAIN ROUTINE: Fill

46

PC BLOCK #

Count in from PC # 46 Find sector of m_3 of H

Skip in from PC # -----

Explanation:

Record (E) into main memory or buffer register (J). If recording into memory channels 00-17, the complementary functions E_5R_1 and E'_5R_1 are used.

Clear E.

Set K true at the end of the word for the next PC number (50).

Set R_1 false at the end of P_0O_{13}.

$P_0O_{13} \longleftrightarrow P_2O_{13}$

Count out to PC # 50 Add 1 to m_3 of H

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
H_0	←	—	—	—	—	—	—	H_2	—	—	—	—	—	→	7A-1
J_0	←	—	—	—	—	$E_5L_5 + J_2L'_5$	—	—	—	—	—	—	—	→	8A-4
K	X	P_2O_{13} X +	P_2O_{13}												9C-5
$_0Y_1$	1														19B-1
R_T	←	—	—	—	—	E_4R_1	—	—	—	—	—	—	—	→	P_2O_{13} of blk 46
R_F	←	—	—	—	—	E'_4R_1	—	—	—	—	—	—	—	→	

Count out with K true

P_0O_{13}

BLOCK FUNCTION: Record (E) into main memory or buffer

MAIN ROUTINE: Fill

47

PC BLOCK #

I M_1 M_2 M_3

Count in from PC # 47 Record (E) into memory or buffer

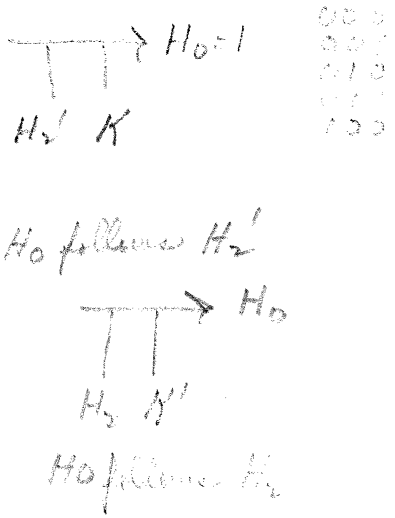
Skip in from PC # ----

Explanation:

J recirculates.

K was true entering block so that equation for H_0 and $_0k$ result in adding 1 to m_3 (0_{0-3}) of H.

K is set true, leaving the block for skip to PC 40.



Count out to PC # ----

Skip out to PC # 40 Reset A_1-A_5

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
															Unit address
H_0	←				H_2						← $KH_2'+K'H_2$			→	7A-2
J_0	←							J_2							8A-5
k	111														9B-2
$_0k$											←	H_2'		→	9A-5

BLOCK FUNCTION: Add 1 to m_3 of H

MAIN ROUTINE: Fill

50

PC BLOCK #

Count in from PC # ----

Skip in from PC # 2 Test on (A_4)

Explanation:

H and J recirculate.

K is set false in PC 54 and remains false to count to PC 57, where it is set true to skip to PC 40. Shift (E) left one octal or decimal digit in each of the blocks 54, 55, 56 and 57.

F key = 1 zero + 3 octal shifts

45 zeros + 5

Count out to PC # ----

Skip out to PC # 40 Reset A_1-A_5

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	←	←	←	←	←	←	A_1	→	→	→	→	→	→	6A-2
H_0	←	←	←	←	←	←	←	H_2	→	→	→	→	→	→	7A-1
J_0	←	←	←	←	←	←	←	J_2	→	→	→	→	→	→	8A-5
k								(57 only)				111			9B-2
$_0k$				54 only				(57 only)				111			9A-4
a_1								A_2							D-8
$_0a_1$								A_2'							D-4
a_2								A_3							C-8
$_0a_2$				octal shift				A_3'	decimal shift						C-4 (12)
a_3						E_5	A_6' +	A_4	A_6						B-7
$_0a_3$						E_5'	A_6' +	A_4'	A_6						B-3
a_4								E_5							A-9
$_0a_4$								E_5'							A-4

BLOCK FUNCTION: Shift (E) left four digits / Shift (E) four digits

MAIN ROUTINE:

54-57

PC BLOCK #

Count in from PC # ----

Skip in from PC # 20 Test for Command to

Explanation:

H and J recirculate.

Put a marker 1 in $P_0 0_{12}$ of E for counting shifts in Block 65.

Test (E) for overflow and set $A_1 = 1$ if there is an overflow -- that is, if $(E_5) = 1$ at $P_0 0_{12}$.

Count out to PC # ----

Skip out to PC # 65 Shift Control Number)

BLOCK / WORD-TIME / PROPOSITION CHART

	0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0			1													6A2
H_0		←						H_2							→	7A1
J_0		←						J_2							→	8A5
a_1			E_5													12D8

BLOCK FUNCTION: Set up for to

MAIN ROUTINE Test for overflow

60

PC BLOCK #

Count in from PC # ----

Skip in from PC # 400 Test for Test Switch

Explanation:

J recirculates.

Set A_1 =1 if a Test Switch is set corresponding to the coding of a binary digit of H (in 0_9 or $P_0 0_{10}$). $P_2 0_{10}$ sets A_1 true (A_1 =1) if a tape unit (TDM) is searching.

Return Control Number (m_2 of G) to H.

Put marker 1 in $P_0 0_{12}$ of E for shift count in Block 65.

⑤ at $P_0 O_9$ time if sw 2100 is on + bit at this time ($P_0 O_9$) turn on A_1

Count out to PC # ----

Skip out to PC # 65 Shift Control Number

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0		1													6A2
H_0	←		H_2					G_5				H_2		→	7A4
J_0	←						J_2							→	8A5
a_1				① ② ③ ④ ⑤											12D5
						24 OD									
				①	=	H_2	(TDM B.S.)								
				②	=	H_2	(SW 2100)								
				③	=	H_2	(SW 2040)								
				④	=	H_2	(SW 2020)								
				⑤	=	H_2	(SW 2010)								

BLOCK FUNCTION: Set up for Control Number Shift

MAIN ROUTINE: Test Switch or Tape Search

61

PC BLOCK #

Count in from PC # ----

Skip in from PC # 23 Test for Test Magnitude

Explanation:

H and J recirculate.

Set A_1 true if $|(E)| > |(F)|$, that is, the magnitude of 0_{0-11} of E exceeds the magnitude of 0_{0-11} of F.

Put a marker 1 in $P_0 0_{12}$ of E for count in Block 65.

E > F set A_1' false

E 111
F 000

111
101

Count out to PC # ----

Skip out to PC # 65 Shift Control Number

BLOCK / WORD-TIME / PROPOSITION CHART

0 / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
F_0		clearing													
E_0		1		clear E except $P_0 0_{12}$											6A2
H_0	←							H_2						→	7A1
J_0	←							J_2						→	8A5
a_1			←			E_5	F_2'							→	12D8
${}_0a_1$			←			E_5'	F_2							→	12D3

BLOCK FUNCTION: Set up for tm

MAIN ROUTINE Test Magnitude

63

PC BLOCK #

Count in from PC # ----

Skip in from PC # 24 Test for Test Algebraically

Explanation:

H and J recirculate.

Set A_1 true if (E) > (F). Here positive zero is considered greater than negative zero.

Set marker 1 in $P_0 0_{12}$ of E for counting the shift in PC #65. Equations ① and ② modify the state of A_1 if the respective signs of E and F so indicate.

$E \neq F = A_2$

when E+F are different time on A_2 during magnitude portion come in with K true and leave true to skip out

Count out to PC # ----

Skip out to PC # 65 Shift Control Number

BLOCK / WORD-TIME / PROPOSITION CHART

	0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
				Sign portion $P_{1,2}$												
E_0			1													6A2
H_0		←						H_2							→	7A1
J_0		←						J_2							→	8A5
a_1			①	←				E_5	F_2'						→	12C5
${}_0a_1$			②	←				E_5'	F_2						→	12D3
a_2				←				$E_5 F_2' + E_5' F_2$							→	12C7
			①	$a_1 = A_1' F_2 (E_5' + A_1) P_{1,2}$												
			②	${}_0a_1 = A_1 E_5 P_{1,2}$												

BLOCK FUNCTION: Set up for t a

MAIN ROUTINE: Test Algebraically

64

PC BLOCK #

Count in from PC # ----

Skip in from PC # 60, 61, 63, 64

Explanation:

J recirculates.

If A_1 has been set false in PC's 60, 61, 63 or 64, then H recirculates, K is set =1, and the Program Counter counts immediately to PC 66, according to equation ① .

If A_1 has been set true, the operation will stick in PC 65 until the marker previously placed in $P_0 0_{12}$ of E moves to $P_0 0_{13}$. This permits four octal shifts left, with H shifted by $A_3 A_4 A_5$, and E shifted using A_2. After four shifts, E_5 goes true in $P_0 0_{13}$ and K is set false for count to PC 66.

H shifts to get M_3 to M_2 of H

M_3 contains jump

E contains 1 bit in $P_0 0_{12}$

Count out to PC # 66 Dummy

Skip out to PC # Stick in PC 65 until shift is complete

BLOCK / WORD-TIME / PROPOSITION CHART

0 / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
					Shift				P_2 Shift						
E_0	←							A_2						→	6A2
H_0	←					A_1'	A_5	+	A_1'	H_2				→	7A2
J_0	←							J_2						→	8A5
$_0k$	①														9A5
a_2	←							E_5						→	12C9
$_0a_2$	←							E_5'						→	12C4
a_3	←							H_2						→	12B8
$_0a_3$	←							H_2'						→	12B4
a_4	←							A_3						→	12A8
$_0a_4$	←							A_3'						→	12A4
a_5	←							A_4						→	18D4
$_0a_5$	←							A_4'						→	18C4
			①	$_0k = E_5 + A_1'$ count # of shifts until bit is on $P_0 0_{13}$ and $P_0 0_{12}$											

SKIP IN K time

BLOCK FUNCTION: Shift control number

MAIN ROUTINE Test routines to, ts, tm, ta

65

PC BLOCK #

Count in from PC # 65 Shift Control Number

Skip in from PC # -----

Explanation:

Whenever it is necessary to skip after a block where sticking may occur, a dummy block must be inserted since the Program Counter can have only two pairs of choices at any block. These are:

(1) To stick or count.

(2) To skip or count.

There is no such choice as to skip or stick.

Count out to PC # ----

Skip out to PC # 6 Find channel of m_2 of H

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
H_0	←	—	—	—	—	—	—	H_2	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	—	J_2	—	—	—	—	—	→	8A5
k												///			9B2

BLOCK FUNCTION: Dummy
Test routines to, ts, tm, ta

MAIN ROUTINE

66

PC BLOCK #

Count in from PC # ----

Skip in from PC # 663 Test for End of wt (Write Tape)

Explanation:

Set A_7 false.

Put (m_2) of G, the control number, back into m_2 of H.

Count out to PC #----

Skip out to PC # 6 New Command - Select channel of m_2 of H.

BLOCK / WORD-TIME / PROPOSITION CHART

O	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT
P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	LOCATION
H_0	←		H_2					G_5		→	←		H_2	→	7A4
J_0	←						J_2							→	8A5
$_0a_7$														111	18A3

BLOCK FUNCTION: Set A_7 false. Put control number back into H.

MAIN ROUTINE: Write Tape (Magnetic)

67

PC BLOCK #

Count in from PC # ----

Skip in from PC # 25 Extract Command Test

Explanation:

E, F, H and J recirculate.

Set K false in 0_2 for count to PC 76.

L_1-L_5 are set up to select channel indicated in m_3 of H.

Count out to PC # 76 Find sector in m_3 of H

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	←	←	←	←	←	←	E_5	→	→	→	→	→	→	6B-2
F_0	←	←	←	←	←	←	←	F_2	→	→	→	→	→	→	7D-3
H_0	←	←	←	←	←	←	←	H_2	→	→	→	→	→	→	7A-1
J_0	←	←	←	←	←	←	←	J_2	→	→	→	→	→	→	8A-5
$_0k$												///			9A-4
l_1												H_2			D1-2
$_0l_1$												H_2'			D1-2
l_2												H_2			D1, C2
$_0l_2$												H_2'			D1, C2
l_3												H_2			10 D1, B2
$_0l_3$												H_2'			D1, B2
l_4											H_2				B1-2
$_0l_4$											H_2'				B1-2
l_5											H_2				B1, A2
$_0l_5$											H_2'				B1, A2

BLOCK FUNCTION: Find channel in m_3 of H
Extract

MAIN ROUTINE:

75

PC BLOCK #

Count in from PC # 75 Find channel in m_3 of H

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

K is set false at $P_2 0_{13}$ and goes true if digits in 0_{0-1} of M_w do not coincide with 0_{0-1} of H. Only 0_0 of H is tested if look-up is in buffer, according to equation ②.

When M_w and H coincide, K is left false, resulting in count to PC 77.

Count out to PC # 77 Read m_3 into G

Skip out to PC # Stick in PC 76 until sector is found

BLOCK / WORD-TIME / PROPOSITION CHART

	0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0		←							E_5						→	6B-2
F_0		←							F_2						→	7D-3
H_0		←							H_2						→	7A-1
J_0		←							J_2						→	8A-5
k														← ①	①→ ②→	9D-5
$_0k$	X'															9A-4
			①	$k = (M_w H_2' + M_w' H_2) L_5' \Phi_{0-1}$												
			②	$k = (M_w H_2' + M_w' H_2) L_5 L_4' 0_0$												

BLOCK FUNCTION: Find sector in m_3 of H

MAIN ROUTINE: Extract

76

PC BLOCK #

Count in from PC # 76 Find sector in m_3 of H

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

G_0 is connected for one word-time either to main memory (ML'_5), to the buffer ($J_2L_5L'_4L'_1$), or to G_5.

Count out to PC # 100 Extract

Skip out to PC # -----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B-2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D-3
G_0	←	—	$ML'_5+J_2L_5L'_4$				$L'_1+G_5L_5L_4$							→	7C-3
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A-1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A-5

BLOCK FUNCTION: Read(m_3) into G

MAIN ROUTINE Extract

77

PC BLOCK #

Count in from PC # 77 Read m_3 into G_0

Skip in from PC # ----

Explanation:

Set K true in 0_2 to skip to 134 at end of word.

(E) is extracted into G, according to (F), and the result is placed in E.

Wherever a 0 exists in F, the (G) are transferred into E.

Wherever a 1 exists in F, the (E) are not changed.

Count out to PC # ----

Skip out to PC # 134 Select channel of m_3

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	
E_o	←					$E_5 F_2 + G_5 F_2'$									→	6B-3
H_o	←						H_2								→	7A-1
J_o	←						J_2								→	8A-5
k													///			9B-2

BLOCK FUNCTION: Extract

MAIN ROUTINE: Extract

100

PC BLOCK #

Count in from PC # ----

Skip in from PC # 123 Test for Complement

Explanation:

If A_1 is true, (E) are a negative number (see PC 122) and all zeros in E are changed to ones and vice versa.

Similarly, if A_2 is true, (F) are a negative number and the zeros and ones in F are reversed.

K remains true for skip to PC 124.

Count out to PC # ----

Skip out to PC # 124 Add

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
	Return Insertion														
E_o	← E_5	←					E'_5	A_1	+	E_5	A'_1			→	6A5
F_o		←					F'_2	A_2	+	F_2	A'_2			→	7D3
G_o	←						G_5							→	7B2
H_o	←						H_2							→	7A1
J_o	←						J_2							→	8A5

BLOCK FUNCTION: Ones Complement

MAIN ROUTINE: Add

102

PC BLOCK #

Count in from PC # ----

Skip in from PC # 26 Scale Factor Command Test

Explanation:

F → G and F clears

E, H and J recirculate.

Set K false to count to PC 110.

Set A_{10} true (used in PC 136) to indicate a double-word put-away.

Read (F) into G, clearing F. F contained the number; E contains the original exponent.

Count out to PC # 110 Test for .4

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B-2
G_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D-3
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A-1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A-5
ok												///			9A-4
a_{10}	←	—	—	—	—	—							—	→	8A-1

BLOCK FUNCTION: Set-up

MAIN ROUTINE Scale Factor

107

PC BLOCK #

Count in from PC # 107 Scale Factor Set-Up

Skip in from PC # 111 Shift and Add

Explanation:

E, F, G, H and J recirculate.

Test for 1 in $P_2 0_{11}$ of G (magnitude .4) and when true, set K true to skip to PC 123 (Add Routine).

* If $P_2 0_{11}$ of G is still zero, after 64 (100 octal) times through Blocks 110-111, $P_0 0_2$ of F will contain a 1, setting K true (according to equation ①, $K = F_2 P_0 0_2$) and causing skip to PC 123.

Set A_1 true if (E) is negative

Set A_2 true for next block.

store sign of E line

Count out to PC # 111 Shift and Add

Skip out to PC # 123 Test for Complement

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←							E_5						→	6B-2
F_0	←							F_2						→	7D-3
G_0	←							G_5						→	7B-2
H_0	←							H_2						→	7A-1
J_0	←							J_2						→	8A-5
k			②									①	←		9C-2
a_2														111	12C-7
a_1		E_5													
				①	$K = F_2 P_0 0_2$ *										
			or	②	$K = G_5 P_2 0_{11}$										

limiter if G line has all zeros

shift left limiter

BLOCK FUNCTION: Test for .4

MAIN ROUTINE: Scale Factor

110

PC BLOCK #

Count in from PC # 110 Test for .4

Skip in from PC # ----

Explanation:

E, H and J recirculate.

K is set true in 0_2 for skip, and then false at $P_2 0_{13}$ to enter PC 110 false.

Add 1 to F, using A_2 as the carry flip-flop.

Stuff minus sign in F.

Shift magnitude of G left, using A_3, while 0_{12-13} of G recirculates.

E retains original exponent.

F contains number of shifts (see PC 110).

G contains the shifted number.

shift G one bit left and add one into F

Count out to PC # ----

Skip out to PC # 110 Test for .4

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	
E_0		←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B-2
F_0			1	←	—	—	A_2	F_2'	+	A_2'	F_2	—	—	—	→	7D-4
G_0		← G_5	→	←	—	—	—	—	A_3	—	—	—	—	—	→	7C-2
H_0		←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A-1
J_0		←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A-5
k													///			9B-2
$_0k$		X'														9A-4
a_3				←	—	—	—	G_5	—	—	—	—	—	—	→	D-6
$_0a_3$				←	—	—	—	G_5'	—	—	—	—	—	—	→	12 D-2
$_0a_2$				←	—	—	—	F_2'	—	—	—	—	—	—	→	C-3

Count to Adder
Shift left G line

BLOCK FUNCTION: Shift and Add

MAIN ROUTINE: Scale Factor

111

PC BLOCK #

Count in from PC # ----

Skip in from PC # 33 Test for dd

Explanation:

E, F, H and J recirculate.

Set A_{10} true for two-word put-away.

Set K false.

Count out to PC # 114 Set up for Divide Routine

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	
E_0		←	—	—	—	—	—	—	E_5	—	—	—	—	—	→	6B-2
F_0		←	—	—	—	—	—	—	F_2	—	—	—	—	—	→	7D-3
H_0		←	—	—	—	—	—	—	H_2	—	—	—	—	—	→	7A-1
J_0		←	—	—	—	—	—	—	J_2	—	—	—	—	—	→	8A-5
ok													///			9A-4
a_{10}		←	—	—	—	—	—	—	1	—	—	—	—	—	→	18A-4

BLOCK FUNCTION: Marker for two-word put-away

MAIN ROUTINE: Divide with Remainder

113

PC BLOCK #

Count in from PC # 113 Marker for Divide with Remainder (dd)

Skip in from PC # 34 Test for Divide and Round Off (dr)

Explanation:

Prepare to divide. Retain magnitudes of dividend in E, divisor in F.

Put marker 1 in $P_0 0_1$ of G to indicate end of division in PC 115. H and J recirculate.

Set K false for count to PC 115.

Set A_1 true if E < F; leave A_1 false if E ≥ F.

Set A_2 true to act as carry flip-flop.

In PC 115, A_3 will shift (G). A_4 retains the sign of E, and A_5 will shift (E).

A_6 retains the sign of F. A_7 is set true if there is no overflow, that is if E < F.

A_8 will be true in PC 115 if E < F.

A_{10} has been set true for division with remainder, if required.

Count out to PC # 115 Divide

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0			←	—	—	—	E_5	—	—	—	—	—	—	→	6A-2
F_0			←	—	—	—	F_2	—	—	—	—	—	—	→	7D-4
G_0													1		7B-1
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A-1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A-5
$_0k$												///			9A-4
a_1			←	—	—	—	E_5'	F_2	—	—	—	—	—	→	12C-7
$_0a_1$			←	—	—	—	E_5	F_2'	—	—	—	—	—	→	12C-2
a_2														///	12C-7
a_4		E_5													12A-8
a_6		F_2													18C-3
a_7			←	—	—	—	E_5'	F_2	—	—	—	—	—	→	18B-4
$_0a_7$			←	—	—	—	E_5	F_2'	—	—	—	—	—	→	18A-4

BLOCK FUNCTION: Set up for Divide Routine

MAIN ROUTINE: Divide

114

PC BLOCK #

Count in from PC # 115 Set up to Divide

Skip in from PC # ----

Explanation:

If A_1 is true (E < F), do not subtract (F) from (E) and put a zero in $P_0 0_0$ of G (the quotient register). Shift (E) left through A_5 and (G) left through A_3. Use A_8 to compare (F) with shifted (E). Continue until (E) ⩾ (F).

If A_1 is false (E ⩾ F), subtract (F) from (E) using A_2 and shift the remainder left, comparing the remainder with (F) using A_8. Record a 1 in G.

At the end of the word time, test for a marker in $P_0 0_{13}$ of G. If no pulse, K is left true and stick in block to complete division. Repeat the block until the marker set in $P_0 0_1$ of G in PC 114 is shifted to $P_0 0_{13}$, that is, for 37 bits.

Also at end of block, set A_1 to follow A_8 which just compare (E) and (F).

At end of division, E contains remainder; G contains quotient.

Count out to PC # 116 Test for Divide and Round Off

Skip out to PC # Stick in PC 115 until Divide Completed

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	—	A_5	—	—	—	—	—	→	6A-2
F_0	←	—	—	—	—	—	—	F_2	—	—	—	—	—	→	7D-3
G	←	—	—	—	—	—	—	A_3	—	—	—	—	—	→ ①	7B-4
H_0	←	—	—	—	—	—	—	H_2	—	—	—	—	—	→	7A-1
J_0	←	—	—	—	—	—	—	J_2	—	—	—	—	—	→	8A-5
ℓ												111			9B-2
$_0\ell$	G_5														9A-3
a_1		←A_8→													12C-8
$_0a_1$		←A_8'→													12C-3
a_2	111		←	—	—	—	E_5	F_2'	—	—	—	—	—	→	12C-8
$_0a_2$			←	—	—	—	E_5'	F_2	—	—	—	—	—	→	12C-3
a_3	←	—	—	—	—	—	—	G_5	—	—	—	—	—	→	12B-9
$_0a_3$	←	—	—	—	—	—	—	G_5'	—	—	—	—	—	→	12B-4
a_5		←E_5	A_1+	$A_1'($	$E_5F_2'A_2$		$+E_5$	$F_2A_2'+$		$E_5'F_2'A_2'$		$+E_5'F_2A_2)$		→	18C2,3
$_0a_5$		←[E_5	A_1+	$A_1'($	E_5	$F_2'A_2$	$+E_5$	F_2	$A_2'+$	E_5'	$F_2'A_2'$	$+E_5'$	F_2	A_2]' →	18C2
a_8		←	—	—	—	—	A_5'	F_2	—	—	—	—	—	→	18B1
$_0a_8$	111	←	—	—	—	—	A_5	F_2'	—	—	—	—	—	→	18A3
			①	$G=$	$A_1'P_0$	O_0									

BLOCK FUNCTION: Divide

MAIN ROUTINE: Divide

115

PC BLOCK #

Count in from PC # 115 Divide

Skip in from PC # ----

Explanation:

Test for Double Precision.

If A_{10} is true, set K true and skip to PC 134. If A_{10} false, K remains false and count to PC 117.

Set A_3 according to A'_1, which will be a 1 if E ≥ F at the end of the 37th word-time in PC 115.

Put a minus sign for the quotient into G, if the signs of the dividend (A_4) and divisor (A_6) were different. See equation ④ and PC 114.

Put the sign of the dividend into E, equation ② .

Reverse (A_7) at $P_2 0_{11}$.

Record an overflow pulse in both the quotient and remainder (① and ③) at $P_0 0_{12}$ if (A_7) = 1.

Count out to PC # 117 Round Off

Skip out to PC # 134 Select channel of m_3 of H

BLOCK / WORD-TIME / PROPOSITION CHART

0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0		⑳ ←	←	←	←	←	E_4	→	→	→	→	→	→	→	6D-4
G_0		⑦⑨ ←	←	←	←	←	G_5	→	→	→	→	→	→	→	7C-4
H_0	←	←	←	←	←	←	H_2	→	→	→	→	→	→	→	7A-1
J_0	←	←	←	←	←	←	J_2	→	→	→	→	→	→	→	8A-5
K	←	←	←	←	←	←	A_{10}	→	→	→	→	→	→	→	9B-3
a_3			A'_1												12B8
a_7			A'_7												18B2
$\circ a_7$			A_7												18A3
				①	$E_0 \neq A_7$	P_0	0_{12}								
				②	$E_0 \neq A_4$	P_1	0_{12}								
				③	$G_0 \neq A_7$	P_0	0_{12}								
				④	$G_0 \neq (A'_4 A_6 + A_4 A'_6)$	P_1	0_{12}								

BLOCK FUNCTION: Test for Double Precision

MAIN ROUTINE: Divide

116

PC BLOCK #

Count in from PC # 116 Test for Double Precision

Skip in from PC # ----

Explanation:

H and J recirculate.

Round off the number in G by adding (A_3) (see PC 116) to G; and put the results in E.

Take the sign of the result from G and place in E. Set K true for skip to PC 134.

Count out to PC # ----

Skip out to PC # 134 Select channel in m_3 of H

BLOCK / WORD-TIME / PROPOSITION CHART

	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
P	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	
E_o	← G_5 →	←	—	—	—	G_5	A'_3	+	G'_5	A_3	—	—	—	→	6D-5
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A-1
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A-5
k												///			9B-2
$_oa_3$			←	—	—	—	—	G'_5	—	—	—	—	—	→	12B2

BLOCK FUNCTION: Round off

MAIN ROUTINE: Divide

117

PC BLOCK #

Count in from PC # ----

Skip in from PC # 21 Test for Subtract

Explanation:

E, G, H, J recirculate.

Reverse sign of F.

Set K false for count to PC 122.

Count out to PC # 122 Test for Number or Command

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←						E_5							→	6B-2
F_o		F'_2	←				F_2							→	7D-2
G_o	←						G_5							→	7B-2
H_o	←						H_2							→	7A-1
J_o	←						J_2							→	8A-5
$_o k$												///			9A-4

BLOCK FUNCTION: Reverse Sign of F

MAIN ROUTINE Subtract

121

PC BLOCK #

Count in from PC # 121 Reverse Sign of F

Skip in from PC # 22 Test for Add

Explanation:

E, F, G, H and J recirculate.

Set K false at the start of the block.

If (E) is a command, then there is a 1 in $P_2 0_{12}$ or P_0, $P_1 0_{13}$. This sets K true for skip to PC 126.

If E contains a number, K stays false for count to PC 123.

Set A_1 true if (E) is negative; A_2 true if (F) is negative.

F=M1

test if cmd or numerical value
(bit in any position during
$P_2 0_{12} + P_0 0_{10} + P_1 0_{13}$)

Count out to PC # 123 Test for Complement

Skip out to PC # 126 Complement F if necessary

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B-2
F_0	←						F_2							→	7D-3
G_0	←						G_5							→	7B-2
H_0	←						H_2							→	7A-1
J_0	←						J_2							→	8A-5
k	E_5														9D-3
$_0k$												///			9A-4
a_1		E_5													12D7
a_2		F_2													12B8

BLOCK FUNCTION: Test for Number or Command

MAIN ROUTINE: Add

122

PC BLOCK #

Count in from PC # 122 Test for Number or Command

Skip in from PC # 110 Test for .4

Explanation:

E, F, G, H and J recirculate.

Set K false if entering from PC 110, and then set K true if (E) or (F) (but not both) are negative. A_1 and A_2 retain the sign information from PC 122.

If K true, skip to PC 102.

If K false, with (E) and (F) of like sign, count to PC 124.

① A_1, A_2 different times K

Count out to PC # 124 Add

Skip out to PC # 102 Ones complement

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←						E_5							→	6B-2
F_o	←						F_2							→	7D-3
G_o	←						G_5							→	7B-2
H_o	←						H_2							→	7A-1
J_o	←						J_2							→	8A-5
k	←①														9C-3
				① $k = (A_1 A_2' + A_1' A_2)_{O_{13}}$											
ok												///			

BLOCK FUNCTION: Test for complement

MAIN ROUTINE: Add

123

PC BLOCK #

Count in from PC # 123 Test for Complement

Skip in from PC # 102 Complement

Explanation:

G, H and J recirculate.

Add (E) to (F) and place the sum in E. Enter the sign and overflow into E by equations ① and ② .

If both numbers positive ($A'_1A'_2$): $E_{0\text{-}11}$ contains magnitude of sum, sign is +, carry into $P_0 0_{12}$ (K true) indicates overflow.

If both numbers negative (A_1A_2): $E_{0\text{-}11}$ contains magnitude of sum, sign is -, carry into $P_0 0_{12}$ (K true) indicates overflow.

If numbers are of unlike sign ($A_1A'_2 + A'_1A_2$)

- Carry into $P_0 0_{12}$ (K true) means $E_{0\text{-}11}$ contains magnitude of sum, sign is +, no overflow. No carry into $P_0 0_{12}$ (K false) means $E_{0\text{-}11}$ contains complement of magnitude of sum, sign is -, no overflow. (Count to PC 125).

If overflow, set A_7 true at $P_0 0_{12}$.

Set K true by equation ③ if signs are alike, for skip to PC 134. Set K true at end of block. If complementing is necessary, K will be true entering PC 124, thus adding 1 to the LSD.

② if overflow — in fill and want to retain it put in M_1 section

Count out to PC # 125 Complement (E)

Skip out to PC # 134 Select channel of m_3 of H

BLOCK / WORD-TIME / PROPOSITION CHART

0 / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
							serial adder								
E_o		①② ←		E_5F_2K		$+E_5$	F'_2	$K'+$	E'_5	$F_2K'+$	E'_5	F'_2	K →		6D-5
G_o	←						G_5							→	7B-2
H_o	←						H_2							→	7A-1
J_o	←						J_2							→	8A-5
k	③ 1		←			E_5	F_2							→	9D-2, C-5
$_0k$			←			E'_5	F'_2							→	9A-4
a_7		④													18B3
				①	$E_o=$	$[A_1$	A_2+	$K'(A_1$	A'_2+	$A'_1A'_2)]$	P_1	0_{12}			sign
				②	$E_o=$	$[E_5$	$+K$	$(A_1A_2+$	$A'_1A'_2)]$	P_0	0_{12}				overflow
				③	$k=$	$(A_1$	A_2+	$A'_1A'_2)$	0_{13}	$+P_2Q_{13}$					skip or count
				④	$a_7=$		K	$(A_1$	A_2+	A'_1	$A'_2)$	P_0Q_{12}			set A_7 if overflow

BLOCK FUNCTION: Add

MAIN ROUTINE: Add

124

PC BLOCK #

Count in from PC # 124 Add

Skip in from PC # ----

Explanation:

G, H, J recirculate.

Complement (E) over 0_{o-11}

by $E = (E_5' K' + E_5 K)\, 0_{o-11}$ and ${}_o k = E_5\, {}_o 0_{o-11}$

Set K true in 0_{13} for skip to PC 134.

Count out to PC # ----

Skip out to PC # 134 Select Channel of m_2 of H

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
															1 if true coming into block
E_o	← E_5 →	←				$E_5' K' + E_5 K$								→	6C-5
G_o	←					G_5								→	7B-2
H_o	←					H_2								→	7A-1
J_o	←					J_2								→	8A-5
k	111														9B-2
${}_o k$			←			E_5								→	9A-3

BLOCK FUNCTION: Complement (E)

MAIN ROUTINE: Add

125

PC BLOCK #

true complement of number in E

Count in from PC # ----

Skip in from PC # 122 (Test for Number or Command)

Explanation:

E, G, H, J recirculate.

Complement (F) if its sign is negative.

Set K false at start of block 0_2, for count to PC 127.

Set K according to (A_2) at $P_2 0_{13}$.

Count out to PC # 127 (Add)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B-2
F_0			←	—	—	F_2'	A_2+	$F_2 A_2'$	—	—	—	—	—	→	7D-3
G_0	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B-2
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A-1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A-5
k	A_2														9C-5
$_0k$												///			9A-4

BLOCK FUNCTION: Complement F if necessary

MAIN ROUTINE: Add

126

PC BLOCK #

Count in from PC # 126 (Complement (F))

Skip in from PC # ----

Explanation:

Add (F) to (E).

Count out to PC # ----

Skip out to PC # 134

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	
E_0		← E_5 →		←	← E_5	F_2K	$+E_5$	F_2'	$K'+$	E_5'	$F_2K'+E_5'$		F_2'	K	→	6A-5
G_0		←						G_5							→	7B-2
H_0		←						H_2							→	7A-1
J_0		←						J_2							→	8A-5
k		111		←				E_5	F_2						→	9C-2
$_0k$				←				E_5'	F_2'						→	9A-4

serial adder →

BLOCK FUNCTION: Add (F) to Command

MAIN ROUTINE: Add

127

PC BLOCK #

Count in from PC # ----

Skip in from PC # 30 (Test for sl)

Explanation:

Test for end of shift process (K left true after subtracting 1 from 0_{0-11} of F).

Set A_1 false for shifting in next block.

Store sign of (F) in A_2.

a_2 true = shift R

a_2' true = shift L

Count out to PC # 131 (Shift (E))

Skip out to PC # 134

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B-2
F_0	← F_2	→	←	—	—	F_2 K'	+ F_2'	K	—	—	—	—	—	→	7D-2
G_0	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B-2
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A-1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A-5
$_0k$			←	—	—	—	F_2	—	—	—	—	—	—	→	9A-2
$_0a_1$														///	12 D-1
a_2		F_2													12 B-8
					a_2	= F_2	$P_1 O_{12}$								

BLOCK FUNCTION: Test for end of shifting

MAIN ROUTINE: Shift entire word

130

PC BLOCK #

Count in from PC # 130 (Test for end of shift)

Skip in from PC # ----

Explanation:

Shift (E) right if $(A_2) = 1$, left if $(A_2) = 0$, using A_1 as shifting FF in latter case.

Set K true for skip to PC 130.

Count out to PC # ----

Skip out to PC # 130

BLOCK / WORD-TIME / PROPOSITION CHART

0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
									Leave bit at right end in E_4						
E_0	①				A_2'	A_1	+ A_2	E_4	✓					→	6A-3
F_0	←						F_2							→	7D-3
G_0	←						G_5							→	7B-2
H_0	←						H_2							→	7A-1
J_0	←						J_2							→	8A-5
k												///			9B-2
a_1	←						E_5							→	12D-9
$_0a_1$	←						E_5'							→	12D-4

$E_0 = (A_2'A_1 + A_2E_4)Q_{0-12}P_{0,1}Q_{13}$

~~② $E_0 = A_2'A_1P_{0,1}Q_{13}$~~

① erase head fills in zero

BLOCK FUNCTION: Shift

MAIN ROUTINE: Shift over entire word

131

PC BLOCK #

Count in from PC # ----

Skip in from PC # 27 (Test for sm)

Explanation:

Test for end of shift process (K left true after subtracting 1 from 0_{0-11} of F).

Set A_1 false for shifting in next block.

Store sign of (F) in A_2. shift left or right
right = − sign

K remain true to end shifting
and skip for putaway (134)

to keep octal # but shift = each shift is multiple of 3

Count out to PC # 133 (Shift (E))

Skip out to PC # 134

BLOCK / WORD-TIME / PROPOSITION CHART

0 / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0 ←							E_5							→	
E_0	~~E_5~~	10				F_5								→	6C-3
F_0	← F_2 →					$F_2K' + F_2'K$								→	7D-2
G_0	←					G_5								→	7B-2
H_0	←					H_2								→	7A-1
J_0	←					J_2								→	8A-5
$_0k$			←			F_2								→	9A-2
$_0a_1$														111	12D-1
a_2		F_2													12B-8
		~~0 E_5 A_1 A_0 a_2~~													

BLOCK FUNCTION: Subtract 1 from F

MAIN ROUTINE: Command: Shift Magnitude

132

PC BLOCK #

Count in from PC # 132 (Subtract 1 from F)

Skip in from PC # ----

Explanation:

A_2' shift left ←

A_2 shift right →

Same as PC 131 except that:

1. E recirculates during $0_{12\text{-}13}$.
2. The equation for shifting (E) right (A_2E_4) extends only through $P_1\,0_{11}$ in order to keep the overflow pulse, if any, out of $0_{0\text{-}11}$ of E.
3. Retain any overflow in A_3 for use in PC 132.

check 392 thru 43L taken out of mod 3

shift left E_0 follows A_1 if A_2'

shift right E_0 follow nothing if A_2 true

Count out to PC # ----

Skip out to PC # 132

BLOCK / WORD-TIME / PROPOSITION CHART

0 / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	← E_5 →		① ←		A_2'	A_1	+	A_2	E_4					→	6B-4,5
F_0	←						F_2							→	7D-3
G_0	←						G_5							→	7B-2
H_0	←						H_2							→	7A-1
J_0	←						J_2							→	8A-5
k												///			9B-2
a_1			←					E_5						→	12D-7
$_0a_1$			←					E_5'						→	12C-4
a_3		②													12B-7
			①	$E_0 = A_2'$	A_1	P_2	O_{11}								
			②	~~$a_3 = A_2'$~~	~~A_1~~	~~P_0~~	~~O_{12}~~								

BLOCK FUNCTION: Shift

MAIN ROUTINE: Shift Magnitude

133

PC BLOCK #

Count in from PC # ----

Skip in from PC # 100, 116, 117, 124, 125, 127, 130, 132, 176 or 177

Explanation:

Skip in from all arithmetic operations to record.

E, G, H and J recirculate.

K is set false in 0_2 for count to PC 135.

Set the channel selector flip-flops L_{1-5} to select channel indicated in m_3 of H using 0_2 through $P_1 0_3$.

Count out to PC # 135 (Find Sector of m_3)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B-2
G_0	←						G_5							→	7B-2
H_0	←						H_2							→	7A-1
J_0	←						J_2							→	8A-5
$0k$												///			9A-4
l_1												H_2			D1-2
$0l_1$												H_2'			D1-2
l_2												H_2			D1, C2
$0l_2$												H_2'			D1, C2
l_3												H_2			D1, B2
$0l_3$												H_2'			10 D1, B2
l_4											H_1				B1-2
$0l_4$											H_1'				B1-2
l_5											H_2				B1, A2
$0l_5$											H_2'				B1, A2
$0a_1$														///	

BLOCK FUNCTION: Find channel in m_3 of H

MAIN ROUTINE: Put-away

134

PC BLOCK #

Count in from PC # 134

Skip in from PC # ---

Explanation:

E, G, H and J recirculate.

K enters false and is reset false at $P_2 0_{13}$. K goes true if digits in 0_{0-1} of M_w do not coincide with 0_{0-1} of H. Only 0_0 of H is tested if look-up is in buffer, as shown by $L_5 L'_4$ of equation ② .

When M_w and H digits coincide, K is left false, resulting in count to PC 136.

Set record flip-flop R_1 true if in channels 0-17 when sector is found, according to $K' L'_5$ of equation ③.

Count out to PC # 136

Skip out to PC # Stick in PC 135 until sector is found

BLOCK / WORD-TIME / PROPOSITION CHART

0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B2
G_0	←						G_5							→	7B-2
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A-5
k													← ①	→ ②	9D-5
$_0k$	X'														9A-4
[redacted]	③														19B-1
r_1	③														19B-1
				①	$k = (M_w$	$H'_2 +$	M'_w	$H_2)$	L'_5	0_{0-1}					
			and	②	$k = (M_w$	$H'_2 +$	M'_w	$H_2)$	L_5	L'_4	0_0				
				③	$r_1 = K' L'_5$	[redacted]									
							$P_1 0_{13}$								

BLOCK FUNCTION: Find sector in m_3 of H

MAIN ROUTINE: Put-away

135

PC BLOCK #

Count in from PC # 135 (Find sector of m_3)

Skip in from PC # ----

Explanation:

(E) go into memory; (G) go into E. double pulse gateway

G and H recirculate.

If L_5 is true, the (E) is recorded into the buffer.

K is set true, to skip, if $(A_{10}) = 0$.

Record flip-Flop R_1 is re-set according to the conditions of equation (1)

${}_0r_1 = R_1 A'_{10} P_1 0_{13}$ if only one word is to be recorded.

A_{10} is true for double-length word and R_1 remains true.

$Rt = E_5 R_1 + J_2 R_2$

$Rf = E'_5 R_1 + J'_2 R_2$

(Rt and Rf are the memory record propositions).

$P_0 0_{13}$

A10 double length pulse gateway

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						G_5							→	7C-2
G_0	←						G_5							→	7B-2
H_0	←						H_2							→	7A-1
J_0	←				E_5	L_5	+	J_2	L'_5					→	8A-4
k	←						A'_{10}							→	9B-4
${}_0r_1$	①														9B-1
			①	${}_0r_1$		A'_{10}	$P_0 0_{13}$								
R_T	←					$E_4 R_1$								→	$P_2 0_{13}$ ≠135
R_F	←					$E_4 R_1$								→	

Count out to PC # 137 (Record (E) into Memory)

Skip out to PC # 6 (Next Command - select channel in m_2)

BLOCK FUNCTION: Record (E) into M

MAIN ROUTINE Put-away

136

PC BLOCK #

Count in from PC # 136 (Record (E) into memory)

Skip in from PC # ----

Explanation:

PC 137 is used only for double-length put-away, based on the setting of A_{10} (See PC 136).

G and H recirculate.

If L_5 is true, the (E) is recorded into the buffer.

K is set true in 0_2 for skip to PC 6.

$Rt = E_5R_1 + J_2R_2$

$Rf = E_5'R_1 + J_2'R_2$

LSH pertaining cell M_3

MSH in next physical cell

Count out to PC # ----

Skip out to PC # 6 (Next Command - Select channel in m_2)

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	
G_0		←						G_5							→	7B-2
H_0		←						H_2							→	7A-1
J_0		←				E_5	L_5	+	J_2	L_5'					→	8A-4
k													///			9B-2
or_1		①														19B-1
						②	or_1	=	(13-2)	P_3	Q_{13}					
R_T		K	←					E_4	R_1	←					→	
R_F		K	←					E_4'	R_1	←					→	

BLOCK FUNCTION: Record (E) into Memory

MAIN ROUTINE: Put-away

137

PC BLOCK #

Count in from PC # ----

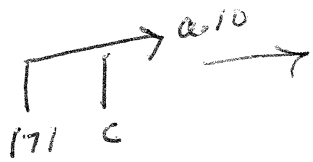
Skip in from PC # 31 (Test: Multiply double length)

Explanation:

E, F, H and J recirculate.

Set K false for count to PC 172.

Set A_{10} true to indicate two-word put-away.



BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←						E_5							→	6B-2
F_o	←						F_2							→	7D-3
H_o	←						H_2							→	7A1
J_o	←						J_2							→	8A-5
$_ok$												///			9A-4
a_{10}	←						1							→	18A-1

Count out to PC # 172 (Set up)

Skip out to PC # ----

BLOCK FUNCTION: Marker for double length

MAIN ROUTINE: Multiply double length

171

PC BLOCK #

Count in from PC # 171 (Marker for md)

Skip in from PC # 32 (Test: Multiply round off)

Explanation:

J recirculates.

Put the sign of (E) in A_4; sign of (F) in A_6.

Multiplier and least significant half of partial product (p.p.) is in E, multiplicand in F; most significant half of p.p. in G.

Clear G and 0_{12-13} of E and F.

Put 3 in 0_{12} of H to count operations.

A_1 is decision flip-flop; A_2 is carry FF.

Use E_4 to set A_1 for PC 173.

Set K false for count to PC 173.

It stores the LS bit of the multiplier in order to indicate whether or not the multiplicand is to be added into the partial product during the next word-time.

Count out to PC # 173 (Multiply)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0			←	—	—	—	E_5	—	—	—	—	—	—	→	6A-2
F_0			←	—	—	—	F_2	—	—	—	—	—	—	→	7D-4
G_0	←	—	—	—	—	—	0	—	—	—	—	—	—	→	
H_0		11	←	—	—	—	H_4	—	—	—	—	—	—	→	7A-2
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A-5
$_0K$												111			9A-4
a_1	E_4														12D8
$_0a_1$	E_4'														12C3
a_4		E_5													12A8
a_6		F_2													18C3

BLOCK FUNCTION: Set up to multiply

MAIN ROUTINE: Multiply

172

PC BLOCK #

Count in from PC # 172 (Set up to multiply)

Skip in from PC # ----

Explanation:

F and J recirculate.

PC 173:

G accumulates MS half of partial product; the LS bit (dropped) is stored in A_3.

If A_1 true, add (F) to shifted p.p. (G) using A_2 as carry FF.

If A_1 false, do not add (F) to (G), but shift (G).

Shift multiplier (E) right and put LSBD (shifted off) in A_1.

Subtract 1 from $0_{12\text{-}13}$ of H. When $0_{12\text{-}13}$ of H is reduced to zero PC counts to next block (174) and $0_{12\text{-}13}$ fills with 37.

PC 174:

Repeat as in PC 173, except subtract 32 times from H $0_{12\text{-}13}$ until all zeros.

Then K is false for count to PC 175.

* Shift partial product right and store dropped bit in A_3.

Record (A_3) (next MSBD of LS half of partial product) into $P_2 0_{11}$ of E).

Count out to PC # 175 (Last shift)

Skip out to PC # Stick in 173 and 174 until operations complete

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o		②	① ←					E_4						→	6B-5
F_o	←							F_2						→	7D-3
G_o			← G_4	$A_1'+A_1(G_4F_2A_2+G_4F_2'A_2'+G_4'F_2A_2'+G_4'F_2'A_2)$											7C-3
H_o	← ③	→	←					H_2						→	7A-4
J_o	←							J_2						→	8A-5
k	←	H_2 →													9B-5
$_o k$	X'														9A-4
a_1	E_4														12D8
$_o a_1$	E_4'														12C3
a_2			←				G_4	F_2						→	12B-7
$_o a_2$	111		←				G_4'	F_2'						→	12C2
a_3														G_{15}	12B6
$_o a_3$														G_{15}'	12B2
				①	$E_0 = A_3 P_2 O_{11}$										
				②	$E_0 = (A_4' A_6 + A_4 A_6') P_1 O_{12}$										
				③	$H_0 = (H_2 K + H_2' K')(O_{12} \neq P_{0-1} O_{13})$										

BLOCK FUNCTION: Multiply

MAIN ROUTINE: Multiply

173 - 174

PC BLOCK #

Count in from PC # 174 (Multiply)

Skip in from PC # ----

Explanation:

H and J recirculate.

Shift E and G right one binary digit. Put binary digit shifted off G (LSBD of MS half of p.p.) in A_3 and record it in $P_2 0_{11}$ of E.

Put sign of product (determined by A_4 and A_6) in E and G.

K remains false for count to PC 176.

Count out to PC # 176 (Test for Double length Product)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0		②	①					E_4						→	6B-5
G_0		③						G_4						→	7C-2
H_0	←							H_2						→	7A-1
J_0	←							J_2						→	8A-5
a_3														G_5	12B6
$_0a_3$														G_5'	12B2
				①	$E = A_3 P_2 0_{11}$										
				②	$E = (A_4' A_6 + A_4 A_6') P_1 0_{12}$										
				③	$G = (A_4' A_6 + A_4 A_6') P_1 0_{12}$										

BLOCK FUNCTION: Last shift of multiply

MAIN ROUTINE: Multiply

175

PC BLOCK #

Count in from PC # 175 (Last shift - multiply)

Skip in from PC # ----

Explanation:

E, G, H, J recirculate

Set K according to A_{10}, previously set in PC 171 if m.d.

If K true, skip to PC 134

If K false (no d.p.) count to PC 177

Count out to PC # 177 (Round off product)

Skip out to PC # 134 (Select channel of m_3 of H)

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B-2
G_0	←						G_5							→	7B-2
H_0	←						H_2							→	7A-1
J_0	←						J_2							→	8A-5
K	←						A_{10}							→	9B-3

BLOCK FUNCTION: Test for Double-length Product

MAIN ROUTINE: Multiply

176

PC BLOCK #

177: See 117

Count in from PC # 176

Skip in from PC # ----

Explanation:

H and J recirculate.

Round off the number in G by adding (A_3) (see PC 116) to G; and put the results in E.

Take the sign of the result from G and place in E. Set K true for skip to PC 134.

Count out to PC # ----

Skip out to PC # 134

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	
E_0		← G_5	→	←	—	—	G_5	A_3'	+	G_5'	A_3	—	—	—	→	
H_0		←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	
J_0		←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	
ℓ													111			
a_3				←	—	—	—	—	G_5'	—	—	—	—	—	→	

BLOCK FUNCTION: Round off

MAIN ROUTINE: Multiply

177

PC BLOCK #

Count in from PC # ----

Skip in from PC # 244 (Test for Last Digit)

Explanation: G+E clear

F, H, J recirculate.

Set A_1-A_6 to Flex code for "space", that is,

$$\underset{A_6\,A_5\,A_4\,A_3\,A_2\,A_1}{1\,1\,0\,1\,0\,0} = \text{64 (Octal Code)}$$

61 space code

Set K false for count to PC 205.

Note: The octal equivalent for space is 64 inside the computer. This differs from the Flex code for space (04), because the six FF's are paired off with the six columns across the tape as below:

A'_6	A'_5	A'_3	A'_1	A_2	A_4
0	0	0	0	0	0

In this case, then, the A proposition would have the following values:

A'_6	A'_5	A'_3	A'_1	A_2	A_4	
0	0	0	1	0	0	giving Fx 04 (space)

Count out to PC # 205 (Dummy)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D-3
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A-1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A-5
$_0k$												///			9A-4
$_0a_1$														///	12D-1
$_0a_2$														///	12C3
a_3														///	12A7
$_0a_4$														///	12A3
a_5														///	18D1
a_6														///	18C3

BLOCK FUNCTION: Set up for space

MAIN ROUTINE: Print

204

PC BLOCK #

Count in from PC # 204 (Space)

Skip in from PC # ----

Explanation:

F, H and J recirculate.

K remains false for count to PC 206.

Count out to PC # 206 (Actuate Key)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O / P	13 / 210	12 / 210	11 / 210	10 / 210	9 / 210	8 / 210	7 / 210	6 / 210	5 / 210	4 / 210	3 / 210	2 / 210	1 / 210	0 / 210	COMPONENT LOCATION
F_o	←								F_2					→	7D-3
H_o	←								H_2					→	7A-1
J_o	←								J_2					→	8A-5

BLOCK FUNCTION: Dummy

MAIN ROUTINE: Print

205

PC BLOCK #

Count in from PC # 205

Skip in from PC # ----

Explanation:

F, H and J recirculate.

Actuate Flex Key as coded in A_1-A_6.

When "Go Ahead" signal goes false, K remains false for count to next block.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
F_0	←	—	—	—	—	—	—	F_2	—	—	—	—	—	→	7D-3
H_0	←	—	—	—	—	—	—	H_2	—	—	—	—	—	→	7A-1
J_0	←	—	—	—	—	—	—	J_2	—	—	—	—	—	→	8A-5
k	←	① →													9B-5
$_0k$												III			9A-4
				①	K= (Go Ahead) O_{12-13}										

Count out to PC # 207 (Wait for Flex)

Skip out to PC # Stick until "Go Ahead" ends

BLOCK FUNCTION: Actuate Flex Key

MAIN ROUTINE: Print

206

PC BLOCK #

Count in from PC # 206 (Actuate Flex Key)

Skip in from PC # ----

Explanation:

F, H and J recirculate.

Reset A_5 and A_6 false.

Wait in block, with K set true, until "Go Ahead" from Flex sets K false for count to next block.

Count out to PC # 210

Skip out to PC # Stick in block until "Go Ahead" becomes true

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
F_0	←							F_2						→	7D-3
H_0	←							H_2						→	7A-1
J_0	←							J_2						→	8A-5
k												///			9B-2
$_0k$	←					Go	Ahead							→	9A-3
$_0a_5$														///	18C3
$_0a_6$														///	18B3

BLOCK FUNCTION: Wait for Flex to finish printing

MAIN ROUTINE: Print

207

PC BLOCK #

Count in from PC # 207 (Wait for Flex)

Skip in from PC # 236 (Addresses or Numbers)

Explanation: G + E clear

F, H and J recirculate.

Set L_1-L_5 to find m_1 of H.

Then, with K set false, count to PC 211.

Count out to PC # 211 (Select sector of m_1 of H)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
F_o	←						F_2							→	7D3
H_o	←						H_2							→	7A1
J_o	←						J_2							→	8A-5
$_ok$												///			9A-4
l_1				H_2											D2
$_ol_1$				H_2'											D2
l_2				H_2											C2
$_ol_2$				H_2'											C2
l_3				H_2											10 D2 C2
$_ol_3$				H_2'											B2
l_4			H_2												B2
$_ol_4$			H_2'												B2
l_5			H_2												C2
$_ol_5$			H_2'												C2

BLOCK FUNCTION: Select channel to m_1 of H

MAIN ROUTINE: Print

210

PC BLOCK #

Count in from PC # 210 (Select channel of m_1 of H)

Skip in from PC # ----

Explanation:

Similar to PC 12 except that F, H and J recirculate.

K remains false, after sector is found, for count to PC 212.

Count out to PC # 212 (Read (M) into E: test alph. or #)

Skip out to PC # Stick in PC 211 until sector of m_1 found.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
F_0	←							F_2						→	7D3
H_0	←							H_2						→	7A1
J_0	←							J_2						→	8A-5
k					←	①	→								9D-4
						②									
$_0k$	X'														9A-4
				①	$k=(M_w H_2' + M_w' H_2) L_5' O_{8-9}$										
				②	$k=(M_w H_2' + M_w' H_2) L_5 4_4' 4_1' O_8$										

BLOCK FUNCTION: Select sector of m_1 of H

MAIN ROUTINE: Print

211

PC BLOCK #

Count in from PC # 211 (Find sector)

Skip in from PC # ----

Explanation:

F, H, J recirculate.

Read word from memory into E.

Set K true if alphabetical print mode. * 10 *

ϕ count

Count out to PC # 213 (Test for octal or decimal print)

Skip out to PC # 250 (Alphabetical print set up)

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	M	L'_5	$+J_2$	L_5	L'_4	L'_1+	G_5	L_5	L_4				→	6B-5
F_0	←						F_2							→	7D-3
H_0	←						H_2							→	7A-1
J_0	←						J_2							→	8A-5
~~k~~	F_2														9C-2

BLOCK FUNCTION: Fill E from memory

MAIN ROUTINE: Print

212

PC BLOCK #

Count in from PC # 212

Skip in from PC # ----

Explanation:

Recirculate E, F, H and J.

Set K true if 1 in $P_0 0_{12}$ of F (indicator for decimal printing.) 01 03 *

Count out to PC # 214 (Octal print set up)

Skip out to PC # 253 (Decimal print set up)

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	
E_0		←							E_5						→	6B-2
F_0		←							F_2						→	7D-3
H_0		←							H_2						→	7A-1
J_0		←							J_2						→	8A-5
k			F_2													9C-2

BLOCK FUNCTION: Test for Octal or Decimal print

MAIN ROUTINE: Print

213

PC BLOCK #

Count in from PC # 213 (Octal or decimal)

Skip in from PC # 434 (error print set up)

Explanation:

E, F, H and J recirculate.

Put a marker pulse in $P_0 0_{13}$ of G. This marker selects octal digits to be printed and is shifted right after each character.

Set K false for count to PC 215.

Count out to PC # 215 (Set up A_1-A_4).

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B-2
F_o	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D-3
G_o	1														7B-1
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A-1
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A-5
ok												///			9A-4

BLOCK FUNCTION: Marker in G

MAIN ROUTINE: Print

214

PC BLOCK #

Count in from PC # 214 (Marker in G)

Skip in from PC # 222 [Shift (G) right]

Explanation:

E, F, G, H and J recirculate.

Set up A_1-A_4.

Octal numbers are retained in A_1-A_3. A_4 is set = 0. The lone marker pulse in G sets G_5, A_7, and A_8 true in successive pulse times, whereby these flip-flops will select the proper binary digits in E (E_5) to be stored in A_1-A_3.

K is set = 0 for count to PC 216.

Count out to PC # 216 (Set up A_1-A_6)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←						E_5							→	6B-2
F_o	←						F_2							→	7D-3
G_o	←						G_5							→	7B-2
H_o	←						H_2							→	7A-1
J_o	←						J_2							→	8A-5
ok												111			9A-4
a_1	←						G_3	E_5						→	12C7
a_{o1}	←						G_5	E'_5						→	12C3
a_2	←						A_7	E_5						→	12B8
a_{o2}	←						A_7	E'_5						→	12C2
a_3	←						A_8	E_5						→	12A8
a_{o3}	←						A_7	E'_5						→	12B1
a_{o4}														111	12A3
a_7	←						G_5							→	18B3
a_{o7}	←						G'_5							→	18A2
a_8	←						A_7							→	18B1
a_{o8}	←						A'_7							→	18A2

BLOCK FUNCTION: Set up A_1-A_4.

MAIN ROUTINE: Print

215

PC BLOCK #

Count in from PC # 215 (Set up A_{1-4})

Skip in from PC # ----

Explanation:

E, F, G, H and J recirculate.

Set A_1-A_6 in Flexowriter code from logic developed from previous set-up of A_1-A_4.

K remains false for count to next block.

The flip-flops A_1-A_6 are set to conform to the flexowriter code as indicated in the note in block 204.

Count out to PC # 217 (Actuate Flex Key)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←							E_5	—					→	6B-2
F_o	←							F_2	—					→	7D-3
G_o	←							G_5	—					→	7B-2
H_o	←							H_2	—					→	7A-1
J_o	←							J_2	—					→	8A-5
a_1		①			①	$a_1 = A_1' A_2' A_3' A_4 P_o O_{12}$									C5
a_2		②		$a_2 = (A_1' A_2' A_3' A_4' + A_1 A_2' A_3 A_4' + A_1 A_2' A_3' A_4) P_o O_{12}$												B6
${}_o a_2$		③	${}_o a_2 = (A_1' A_2 A_3' A_4' + A_1 A_2 A_3' A_4') P_o O_{12}$												B4
a_3		④	$a_3 = A_1' A_2' A_3' A_4 P_o O_{12}$												12 A7
${}_o a_3$		⑤	${}_o a_3 = A_1 A_2 A_3 A_4' P_o O_{12}$												A4
a_4		⑥	$a_4 = (A_1' A_2' A_3' A_4' + A_1 A_2' A_3' A_4') P_o O_{12}$												A7
${}_o a_4$		⑦	${}_o a_4 = (A_1' A_2' A_3' A_4 + A_1' A_2' A_3 A_4 + A_1 A_2 A_3 A_4) P_o O_{12}$												A4
a_5		⑧	$a_5 = (A_1 A_2' A_3' A_4' + A_1' A_2' A_3 A_4 + A_1 A_2 A_3 A_4) P_o O_{12}$												18 C3
a_6		⑨	$a_6 = (A_1' A_2' A_3' A_4' + A_1 A_2' A_3' A_4' + A_1' A_2' A_3 A_4) P_o O_{12}$												18 C1

BLOCK FUNCTION: Set up A_1 to A_6

MAIN ROUTINE: Print

216

PC BLOCK #

Count in from PC # 216

Skip in from PC # ----

Explanation:

All registers recirculate.

Actuate Flex key as coded in A_1-A_6.

When "Go Ahead" signal goes false, K remains false for count to next block.

go ahead light in home position

turn K on to stick in block until flip has started, then go ahead goes false

Count out to PC # 220

Skip out to PC # ---- Stick in 217

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	
F_0	←						F_2							→	
G_0	←						G_5							→	
H_0	←						H_2							→	
J_0	←						J_2							→	
ok												111			
k	← ①	→													

① = (G_0 Ahead) O_{12-13}

BLOCK FUNCTION: Actuate Flex key

MAIN ROUTINE: Print

217

PC BLOCK #

Count in from PC # 217 (Actuate Flex key)

Skip in from PC # ----

Explanation:

All registers recirculate.

Stick in block until "Go Ahead" signal from Flex goes true.

Reset A_5 and A_6.

flex completed cycle time
after K thru go ahead goes true

Count out to PC # 221

Skip out to PC # Stick until "Go Ahead" goes true

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←						E_5							→	
F_o	←						F_2							→	
G_o	←						G_5							→	
H_o	←						H_2							→	
J_o	←						J_2							→	
K												///			
$_oK$	←				Go	Ahead								→	
$_oA_5$														///	
$_oA_c$														///	

BLOCK FUNCTION: Wait for Flex to finish printing

MAIN ROUTINE: Print

220

PC BLOCK #

Count in from PC # 220 (Wait for Flex)

Skip in from PC # ----

Explanation:

E, F, G, H and J recirculate.

Test in F for last digit to be printed. If not the last digit, K remains false for count to PC 222, which shifts G right and skips back to PC 215 for repetition of print-out.

If the marker pulse is in 00, or if the marker has shifted into coincidence with the "end of word" pulse in 0_{0-11} of F, K is set true for skip to PC 225.

test for stopping bit

Count out to PC # 222 [Shift (G) right]

Skip out to PC # 225 (Set for tab)

BLOCK / WORD-TIME / PROPOSITION CHART

O	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT
P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	LOCATION
E_0	←						E_5							→	6B-2
F_0	←						F_2							→	7D-3
G_0	←						G_5							→	7B-2
H_0	←						H_2							→	7A-1
J_0	←						J_2							→	8A-5
k			←			F_2	G_5							G_5 →	9C-5

test for end

BLOCK FUNCTION: Test for last octal digit to be printed

MAIN ROUTINE: Print Octal

221

PC BLOCK #

Count in from PC # 221 (Test for last digit to be printed).

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Shift (G) to the right one octal digit (three bits).

Set K true for skip.

shift G line at 1 ϕ digit (shift bit)

Count out to PC # ----

Skip out to PC # 215 (Set up A_1-A_4)

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B-2
F_0	←						F_2							→	7D-3
G_0		←					G_4							→	7C-2
H_0	←						H_2							→	7A-1
J_0	←						J_2							→	8A5
k												111			9B-2

BLOCK FUNCTION: Shift (G) right

MAIN ROUTINE: Print

222

PC BLOCK #

Count in from PC # ----

Skip in from PC # 221 (Test for Last Digit)

Explanation:

F, H and J recirculate.

Set A_1-A_6 to Flex tab code 011001 =31. (See note on block 204).

Set K false for count to PC 226.

tab

Count out to PC # 226

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
F_0	←					F_2								→	7D-3
H_0	←					H_2								→	7A-1
J_0	←					J_2								→	8A5
$_0k$												///			9A-4
a_1														///	12D6
$_0a_2$														///	12C3
$_0a_3$														///	12B2
a_4														///	12A7
a_5														///	18D1
$_0a_6$														///	18B3

BLOCK FUNCTION: Set for Flex tab.

MAIN ROUTINE: Print

225

PC BLOCK #

Count in from PC # 225 (Space)

Skip in from PC # ----

Explanation:

F, H and J recirculate.

K remains false for count to PC 227

Count out to PC # 227 (Actuate Key)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
F_0	←	—	—	—	—	—	—	—	F_2	—	—	—	—	→	
H_0	←	—	—	—	—	—	—	—	H_2	—	—	—	—	→	
J_0	←	—	—	—	—	—	—	—	J_2	—	—	—	—	→	

BLOCK FUNCTION: Dummy

MAIN ROUTINE Print

226

PC BLOCK #

Count in from PC # 226

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Actuate Flexowriter Key

Stick in block until GO AHEAD ends.

flex slotted →

Count out to PC # 230

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
OP	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	COMPONENT LOCATION
E_0	←								E_5					→	6B-2
F_0	←								F_2					→	7D-3
H_0	←								H_2					→	7A-1
J_0	←								J_2					→	8A-5
k	←GO	AHEAD→													
ok												///			

BLOCK FUNCTION: Actuate Flex. Key

MAIN ROUTINE Print

227

PC BLOCK #

Count in from PC # 227

Skip in from PC # ----

Explanation:

All registers recirculate.

Stick in block until "Go Ahead" signal from Flex goes true.

Reset A_5 and A_6.

flex home

Count out to PC # 231

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	
F_o	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	
G_o	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	
K												111			
$_oK$	←	—	—	—	Go	Ahead				—	—	—	—	→	
$_oA_5$														111	
$_oA_6$														111	

BLOCK FUNCTION: Wait for Flex to finish printing

MAIN ROUTINE Print

230

PC BLOCK #

Count in from PC # 230 (Wait for Flex to finish printing)

Skip in from PC # 275 (Test for Last Digit)

Explanation:

address of next word to be printed out

F and J recirculate.

Add 1 to m_1 (0_{8-11}) of H, making it the new address of the next word to be printed out.

Subtract 1 from m_3 (0_{0-3}) of H, which gives the number of words yet to be printed.

K is used for adding to and subtracting from H. It enters block false and is set true by $H_2 0_{0-3}$ (until end of printing) for adding 1 to m_1.

Also test $P_2 0_{12}$ of (F) for an error marker. (There will be a pulse here if the print routine was entered from PC 434. If an error, set K true and skip to PC Ø (Zero-rest).

If not, count to PC 232.

bit stuffed in Fat $P_2 0_{13}$ if error print

Error Print Out

Count out to PC # 232 (Test for Last Word)

Skip out to PC # 0 (Rest)

BLOCK / WORD-TIME / PROPOSITION CHART

0 / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
F_0	←						F_2							→	7D-3
H_0			← H_2	$K'+H_2'$	K		→	← H_2		→	← H_2	$K+H_2'$	K'	→	4A-4 unit subtracter
J_0	←						J_2							→	8A5
k		F_2									←	H_2		→	9B1, C3
$_0k$		←		H_2'	→										9A3

new print to be printed

BLOCK FUNCTION: Add 1 to m_1 of H. Subtract 1 to m_3 of H. Test for error print

MAIN ROUTINE: Print

231

PC BLOCK #

Count in from PC # 231 (Add 1 to m_1 of H).

Skip in from PC # ----

Explanation:

E, H and J recirculate.

Test M_3 of H for zero.

If last word has been printed, m_3 of H = 0, K is left false, and PC counts to 233.

If not, set K true and skip to PC 236 and start of print routine.

Count out to PC # 233 (Dummy)

Skip out to PC # 236 (Addresses or Numbers)

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←						E_5							→	6B-2
H_o	←						H_2							→	7A-1
J_o	←						J_2							→	8A5
k											←	H_2		→	9B1

skip not tested

BLOCK FUNCTION: Test for Last Word

MAIN ROUTINE: Print

232

PC BLOCK #

Count in from PC # 232 (Test for Last Word)

Skip in from PC # ----

Explanation:

H and J recirculate.

Set K true for skip to PC6.

Count out to PC #----

Skip out to PC # 6 (Select channel of m_2 of H).

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
H_0	←							H_2						→	7A-1
J_0	←							J_2						→	8A5
k												///			9B2

BLOCK FUNCTION: Dummy

MAIN ROUTINE: Print

233

PC BLOCK #

Count in from PC # ----

Skip in from PC # 36 (Test: print) or 232 (Test for last word)

Explanation: G+E clear

F, H and J recirculate.

Test to see whether octal addresses are to be printed with the numbers, or whether the numbers alone. If pulse is in $P_1 0_{12}$ of F (F contains (m_2) of command), then K is set false (an address is to be printed) and count to PC 237. Otherwise, skip to PC 210. 02 03 *

Test Chart

0_{13} of F	0_{12} of F	Meaning
000	000	Octal Numbers alone
000	001	Decimal Numbers alone
000	010	Octal addresses and octal numbers
000	011	Octal addresses and decimal numbers
001	000	Alphabetical

I

000 000 ↑ Δ address error prt.
↑ ↑
alpha

Count out to PC # 237 (Marker in G)

Skip out to PC # 210 (Select channel in m_1 of H)

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
F_0	←						F_2							→	7D-3
H_0	←						H_2							→	7A-1
J_0	←						J_2							→	8A5
o k		F_2													9A4

BLOCK FUNCTION: Addresses or numbers

MAIN ROUTINE: Print

236

PC BLOCK #

Count in from PC # 236 (Addresses or Numbers)

Skip in from PC # -----

Explanation: H to E

F, H and J recirculate. Prepare to print octal address.

Read (H) into E.

Put marker pulse into $P_0 0_{11}$ of G to indicate digit to be printed.

K remains false for count to PC 240.

Count out to PC # 240 (Set up A_1-A_4)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	
E_0		←						H_2							→	6A-3
F_0		←						F_2							→	7D-3
G_0				1												7B-3
H_0		←						H_2							→	7A-1
J_0		←						J_2							→	8A5

BLOCK FUNCTION: Marker in G

MAIN ROUTINE: Print

237

PC BLOCK #

Count in from PC # 237

Skip in from PC # 245

Explanation:

E, F, G, H and J recirculate.

Set up A_1-A_4.

Octal numbers are retained in A_1-A_3. A_4 is set = 0. The lone marker pulse in G sets G_5, A_7, and A_8 true in successive pulse times, whereby these flip-flops will select the proper binary digits in E (E_5) to be stored in A_1-A_3.

K is set = 0 for count to PC 241

Marker in G

P.O. 11

Count out to PC # 241

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B-2
F_o	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D-3
G_o	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B-2
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A-1
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A-5
ok												///			9A-4
a_1	←	—	—	—	—	—	G_5	E_5	—	—	—	—	—	→	12C7
$_o a_1$	←	—	—	—	—	—	G_5	E_5'	—	—	—	—	—	→	12C3
a_2	←	—	—	—	—	—	A_7	E_5	—	—	—	—	—	→	12B8
$_o a_2$	←	—	—	—	—	—	A_7	E_5'	—	—	—	—	—	→	12C2
a_3	←	—	—	—	—	—	A_8	E_5	—	—	—	—	—	→	12A8
$_o a_3$	←	—	—	—	—	—	A_8	E_5'	—	—	—	—	—	→	12B1
$_o a_4$														///	12A3
a_7	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	18B3
$_o a_7$	←	—	—	—	—	—	G_5'	—	—	—	—	—	—	→	18A2
a_8	←	—	—	—	—	—	A_7	—	—	—	—	—	—	→	18B1
$_o a_8$	←	—	—	—	—	—	A_7'	—	—	—	—	—	—	→	18A2

BLOCK FUNCTION: Set up A_1-A_4.

MAIN ROUTINE: Print

240

PC BLOCK #

Count in from PC # 240

Skip in from PC # ----

Explanation:

E, F, G, H and J recirculate.

Set A_1-A_6 in Flexowriter code from logic developed from previous set-up of A_1-A_4.

K remains false for count to next block.

The flip-flops A_1-A_6 are set to conform to the flexowriter code as indicated in the note in block 204_4.

Count out to PC # 242

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←							E_5						→	6B-2
F_0	←							F_2						→	7D-3
G_0	←							G_5						→	7B-2
H_0	←							H_2						→	7A-1
J_0	←							J_2						→	8A-5
a_1		①			①	$a_1 = A'_1 A'_2 A'_3 A_4 P_0 O_{12}$									C5
a_2		②		$a_2 = (A'_1 A'_2 A'_3 A'_4 + A_1 A_2 A_3 A'_4 + A_1 A'_2 A'_3 A_4) P_0 O_{12}$											B6
${}_0a_2$		③	${}_0a_2 = (A'_1 A_2 A'_3 A'_4 + A_1 A_2 A'_3 A'_4) P_0 O_{12}$												B4
a_3		④	$a_3 = A'_1 A'_2 A'_3 A_4 P_0 O_{12}$												12 A7
${}_0a_3$		⑤	${}_0a_3 = A_1 A_2 A_3 A'_4 P_0 O_{12}$												A4
a_4		⑥	$a_4 = (A'_1 A'_2 A'_3 A'_4 + A_1 A'_2 A'_3 A'_4) P_0 O_{12}$												A7
${}_0a_4$		⑦	${}_0a_4 = (A'_1 A'_2 A'_3 A'_4 + A'_1 A'_2 A_3 A_4 + A_1 A_2 A_3 A_4) P_0 O_{12}$												A4
a_5		⑧	$a_5 = (A_1 A'_2 A'_3 A'_4 + A'_1 A'_2 A_3 A_4 + A_1 A_2 A_3 A_4) P_0 O_{12}$												18 C3
a_6		⑨	$a_6 = (A'_1 A'_2 A'_3 A'_4 + A_1 A'_2 A'_3 A'_4 + A'_1 A'_2 A_3 A_4) P_0 O_{12}$												18 C1

BLOCK FUNCTION: Set up A_1 to A_6

MAIN ROUTINE: Print

241

PC BLOCK #

Count in from PC # 241

Skip in from PC # ----

Explanation:

E, F, G, H and J recirculate.

Actuate Flexowriter Key.

Stick in block until GO AHEAD ends.

When go ahead goes low dont time out K
and count out
Flex has started cycle

Wait for go ahead to go low

Count out to PC # 243

Skip out to PC # ---- Stick in 242

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←							E_5						→	6B-2
F_0	←							F_2						→	7D-3
G_0	←							G_5						→	7B-2
H_0	←							H_2						→	7A-1
J_0	←							J_2						→	8A-5
ok												///			
k	← GO AHEAD →														

BLOCK FUNCTION: Actuate Flex. Key

MAIN ROUTINE : Print

242

PC BLOCK #

Count in from PC # 242

Skip in from PC # ----

Explanation:

All registers recirculate.

Stick in block until "Go Ahead" signal from Flex goes true.

Reset A_5 and A_6.

Solenoid gave it time ok and sent out - TG# is home

Count out to PC # 244

Skip out to PC # ---- Stick in 243

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	
E_o	◄							E_5							►	
F_o	◄							F_2							►	
G_o	◄							G_5							►	
H_o	◄							H_2							►	
J_o	◄							J_2							►	
K													111			
$_oK$	◄					Go	Ahead								►	
$_oA_5$															111	
$_oA_6$															111	

BLOCK FUNCTION: Wait for Flex to finish printing

MAIN ROUTINE: Print

PC BLOCK # 243

Count in from PC # 243 (Wait for Flex)

Skip in from PC # ----

Explanation:

F, G, H and J recirculate.

Test for last digit of address is 0_8. If digit is there, set K true and skip to PC 204.

If not, count to PC 245.

test for done printing address

Count out to PC # 245 Shift (G) right

Skip out to PC # 204 (Space)

BLOCK / WORD-TIME / PROPOSITION CHART

O	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT
P	2	1	0	2	1	0	2	1	0	2	1	0	2	1	0	2	1	0	2	1	0	2	1	0	2	1	0	LOCATION
E_0	←							E_6						→	
F_0	←							F_2						→	7D-3
G_0	←							G_5						→	7B-2
H_0	←							H_2						→	7A-1
J_0	←							J_2						→	8A5
k						G_5									9B2

BLOCK FUNCTION: Test for Last Digit

MAIN ROUTINE: Print

244

PC BLOCK #

Count in from PC #244

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Shift (G) to the right one octal digit (three bits).

Set K true for skip.

Shift G line ★

remove 5 FF ($G_3 G_4 G_5$)

Count out to PC # ----

Skip out to PC # 240

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←						E_5							→	6B-2
F_o	←						F_2							→	7D-3
G_o	←						G_2							→	7C-2
H_o	←						H_2							→	7A-1
J_o	←						J_2							→	8A5
K												111			9B-2

BLOCK FUNCTION: Shift (G) right

MAIN ROUTINE: Print

245

PC BLOCK #

Count in from PC # ----

Skip in from PC # 212 [Read (M) into E: Numbers or alph.]

Explanation:

E, F, H, J recirculate.

Put marker pulse in $P_0 0_{11}$ of G for start of alphabetical print-out.

Count out to PC # ---

Skip out to PC # 270 (Set up A_1-A_3)

BLOCK / WORD-TIME / PROPOSITION CHART

O / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←							E_5						→	6B-2
F_0	←							F_2						→	7D-3
G_0			1												7B-3
H_0	←							H_2						→	7A-1
J_0	←							J_2						→	8A5

BLOCK FUNCTION: Marker in G

MAIN ROUTINE: Print

250

PC BLOCK #

Count in from PC # ---

Skip in from PC # 213 (Octal or Decimal)

Explanation:

E, F, H, J recirculate.

Set a marker pulse in $P_0 0_{12}$ of G for decimal printing.

Set K false for count to PC 254.

Count out to PC # 254 (Set up A_1-A_4)

Skip out to PC # ---

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←							E_5						→	6B-2
F_0	←							F_2						→	7D-3
G_0		1													7B-2
H_0	←							H_2						→	7A-1
J_0	←							J_2						→	8A5
$_0K$												///			9A4

BLOCK FUNCTION: Marker in G

MAIN ROUTINE: Print

253

PC BLOCK #

Count in from PC # 253 (Marker in G)

Skip in from PC # ----

Explanation:

See PC 215 for explanation with these exceptions:

A_9 follows A_8. This puts the fourth bit of E_5 following marker in G into A_4, for decimal print-out.

Count out to PC # 255 [Set up A_1-A_6 (±)]

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 20	2 20	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B-2
F_0	←						F_2							→	7D-3
G_0	←						G_5							→	7B-2
H_0	←						H_2							→	7A-1
J_0	←						J_2							→	8A-5
a_1	←						G_5	E_5	—					→	C7
$_0a_1$	←						G_5	E_5'	—					→	C3
a_2	←						A_7	E_5	—					→	B8
$_0a_2$	←						A_7	E_5'	—					→	12 C2
a_3	←						A_8	E_5	—					→	A8
$_0a_3$	←						A_8	E_5'	—					→	B1
a_4	←						A_9	E_5	—					→	A9
$_0a_4$	←						A_9	E_5'	—					→	A3
a_7	←						G_5							→	B3
$_0a_7$	←						G_5'	—						→	A2
a_8	←						A_7	—						→	B1
$_0a_8$	←						A_7'	—						→	18 A2
a_9	←						A_8	—						→	B1
$_0a_9$	←						A_8'	—						→	A3

BLOCK FUNCTION: Set up A_1-A_4

MAIN ROUTINE: Print Decimal

254

PC BLOCK #

Count in from PC # 254 (Set up A_1-A_4)

Skip in from PC # ----

Explanation:

E, F, G, H and J recirculate.

Set A_1-A_6 in Flex code for +, -, p, n or space from previous configuration of A_1-A_4 set in PC 254.

K stays false for count to PC 256.

0000 ---- +
0001 ---- p (positive # with overflow)
0010 ---- -
0011 ---- n (negative # with overflow)
1100 ---- space

Junction pt code

$A_6' A_5' A_3' A_1' A_2 A_4$ = flex code

Count out to PC # 256 (Actuate Key)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←							E_5							→	6B-2
F_o	←							F_2							→	7D-3
G_o	←							G_5							→	7B2
H_o	←							H_2							→	7A1
J_o	←							J_2							→	8A5
a_1			①		①	$a_1 = A_1' A_2' A_3' A_4' P_0 O_{12}$										C4
${}_0a_1$			②	②	${}_0a_1 = (A_1 A_2' A_3' A_4' + A_1 A_2 A_3' A_4') P_0 O_{12}$											D1
a_2			③		③	$a_2 = A_1' A_2' A_3' A_4' P_0 O_{12}$										B6 (12)
a_3			④		④	$a_3 = A_1 A_2 A_3' A_4' P_0 O_{12}$										A8
a_4			⑤		⑤	$a_4 = A_1 A_2' A_3' A_4' P_0 O_{12}$										A8
${}_0a_4$			⑥		⑥	${}_0a_4 = A_1' A_2' A_3 A_4 P_0 O_{12}$										A3
a_5			⑦		⑦	$a_5 = (A_3' A_4' + A_1' A_2' A_3 A_4) P_0 O_{12}$										C3 (18)
a_6			⑧		⑧	$a_6 = (A_1 A_2' A_3' A_4' + A_1 A_2 A_3' A_4' + A_1' A_2' A_3 A_4) P_0 O_{12}$										C1 (18)

BLOCK FUNCTION: Set up A_1-A_6 (±)

MAIN ROUTINE: Print Decimal

255

PC BLOCK #

Count in from PC # 255

Skip in from PC # ----

Explanation:

E, F, G, H and J recirculate.

Actuate Flexowriter Key.

Stick in block until GO AHEAD ends.

Count out to PC # 257

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←							E_5						→	6B-2
F_0	←							F_2						→	7D-3
G_0	←							G_5						→	7B-2
H_0	←							H_2						→	7A-1
J_0	←							J_2						→	8A-5
k		GO AHEAD													
ok												///			

BLOCK FUNCTION: Actuate Flex. Key

MAIN ROUTINE: Print

256

PC BLOCK #

Count in from PC # 256

Skip in from PC # ----

Explanation:

All registers recirculate.

Stick in block until "Go Ahead" signal from Flex. goes true.

Reset A_5 and A_6.

Count out to PC # 260

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT
P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	LOCATION
E_o	←						E_5							→	
F_o	←						F_2							→	
G_o	←						G_5							→	
H_o	←						H_2							→	
J_o	←						J_2							→	
K												///			
$.K$	←			Go	Ahead									→	
$.A_5$														///	
$.A_6$														///	

BLOCK FUNCTION: Wait for Flex to finish printing

MAIN ROUTINE Print

257

PC BLOCK #

Count in from PC # 257 (Wait for Flex) (See PC 207)

Skip in from PC # ---

Explanation:

E, F, H and J recirculate.

Shift (G) right four bits.

shift G marker one decimal position (4 bits)

Count out to PC # 261 (Set up A_1-A_4)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT
	P	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	LOCATION
E_0		◄	—	—	—	—	—	E_5	—	—	—	—	—	—	►	6B2
F_0		◄	—	—	—	—	—	F_2	—	—	—	—	—	—	►	7D3
G_0				◄	—	—	—	G_1	—	—	—	—	—	—	►	7B3
H_0		◄	—	—	—	—	—	H_2	—	—	—	—	—	—	►	7A1
J_0		◄	—	—	—	—	—	J_2	—	—	—	—	—	—	►	8A5

BLOCK FUNCTION: Shift (G) right

MAIN ROUTINE: Print Decimal

260

PC BLOCK #

Count in from PC # 260 [Shift (G) right]

Skip in from PC # 266 [Shift (G) right]

Explanation:

See PC 215 for explanation with these exceptions:

A_9 follows A_8. This puts the fourth bit of E_5 following marker in G into A_4, for decimal print-out. Also, set K false.

Count out to PC # 262 (Set up A_1-A_6)

Skip out to PC # ---

BLOCK / WORD-TIME / PROPOSITION CHART

	0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0		←						E_5							→	6B2
F_0		←						F_2							→	7D3
G_0		←						G_5							→	7B2
H_0		←						H_2							→	7A1
J_0		←						J_2							→	8A5
$_0k$													111			9A4
a_1		←						G_5	E_5						→	C7
$_0a_1$		←						G_5	E_5'						→	C3
a_2		←						A_7	E_5						→	B8
$_0a_2$		←						A_7	E_5'						→	C2
a_3		←						A_8	E_5						→	A8
$_0a_3$		←						A_8	E_5'						→	B1
a_4		←						A_9	E_5						→	A9
$_0a_4$		←						A_9	E_5'						→	A3
a_7		←						G_5							→	B3
$_0a_7$		←						G_5'							→	A2
a_8		←						A_7							→	B1
$_0a_8$		←						A_7'							→	A2
a_9		←						A_8							→	B1
$_0a_9$		←						A_8'							→	A3

12

10

BLOCK FUNCTION: Set up A_1-A_4

MAIN ROUTINE: Print Decimal

261

PC BLOCK #

Count in from PC # 261

Skip in from PC # ----

Explanation:

E, F, G, H and J recirculate.

Set A_1-A_6 in Flexowriter code from logic developed from previous set-up of A_1-A_4.

K remains false for count to next block.

The flip-flops A_1-A_6 are set to conform to the flexowriter code as indicated in the note in block 204.

Count out to PC # 263

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←							E_5	—					→	6B2
F_o	←							F_2	—					→	7D3
G_o	←							G_5	—					→	7B2
H_o	←							H_2	—					→	7A1
J_o	←							J_2	—					→	8A5
a_1		①		$a_1 = (A_1' A_2' A_3' A_4) P_o O_{12}$											C5
a_2		②		$a_2 = (A_1' A_2' A_3' A_4' + A_1 A_2' A_3 A_4' + A_1 A_2' A_3' A_4) P_o O_{12}$											B6
$_o a_2$		③		$_o a_2 = (A_1' A_2 A_3' A_4' + A_1 A_2 A_3' A_4') P_o O_{12}$											B4
a_3		④		$a_3 = (A_1' A_2' A_3 A_4) P_o O_{12}$											A7
$_o a_3$		⑤		$_o a_3 = (A_1 A_2 A_3 A_4') P_o O_{12}$											12 A4
a_4		⑥		$a_4 = (A_1' A_2' A_3' A_4' + A_1 A_2' A_3' A_4') P_o O_{12}$											A7
$_o a_4$		⑦		$_o a_4 = (A_1' A_2' A_3 A_4 + A_1' A_2' A_3 A_4 + A_1 A_2 A_3 A_4) P_o O_{12}$											A4
a_5		⑧		$a_5 = (A_1 A_2' A_3' A_4' + A_1' A_2' A_3 A_4 + A_1 A_2 A_3 A_4) P_o O_{12}$											18 C3
a_6		⑨		$a_6 = (A_1' A_2' A_3' A_4' + A_1 A_2' A_3' A_4' + A_1 A_2' A_3' A_4 + A_1' A_2' A_3 A_4) P_o O_{12}$											C1

BLOCK FUNCTION: Set up A_1-A_6

MAIN ROUTINE: Print

262

PC BLOCK #

Count in from PC # 262

Skip in from PC # ----

Explanation:

E, F, G, H and J recirculate.

Actuate Flexowriter Key.

Stick in block until GO AHEAD ends.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←							E_s						→	6B-2
F_o	←							F_s						→	7D-3
G_o	←							G_s						→	7B-2
H_o	←							H_s						→	7A-1
J_o	←							J_s						→	8A-5
k	←GO AHEAD→														
ok												///			

Count out to PC # 264

Skip out to PC # ----

BLOCK FUNCTION: Actuate Flex. Key

MAIN ROUTINE: Print

263

PC BLOCK #

Count in from PC # 263

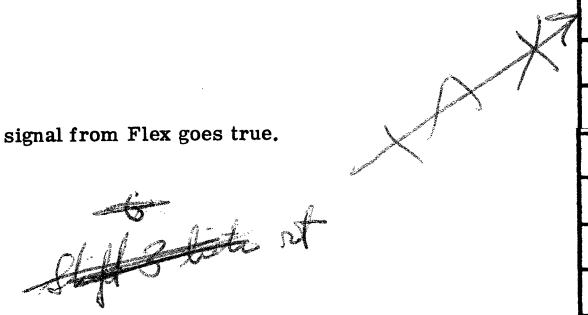
Skip in from PC # ----

Explanation:

All registers recirculate.

Stick in block until "Go Ahead" signal from Flex goes true.

Reset A_5 and A_6.



Count out to PC # 265

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←						E_5							→	
F_o	←						F_2							→	
G_o	←						G_5							→	
H_o	←						H_2							→	
J_o	←						J_2							→	
K												111			
$_oK$	←			Go	Ahead									→	
$_oA_5$														111	
$_oA_6$														111	

BLOCK FUNCTION: Wait for Flex to finish printing

MAIN ROUTINE: Print

264

PC BLOCK #

Count in from PC # 264 (Wait for Flex)

Skip in from PC # ----

Explanation:

E, F, G, H and J recirculate.

Test in F for last digit to be printed. If not the last digit, K remains false for count to PC 222, which shifts G right and skips back to PC 215 for repetition of print-out.

If the marker pulse in in 00, or if the marker has shifted into coincidence with the "end of word" pulse in $0_{0\text{-}11}$ of F, K is set true for skip to PC 225.

test for stopper bit

Count out to PC # 266 [Shift (G) right]

Skip out to PC # 225 (Set for Tab)

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2
H	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k		←	—	—	—	—		F_2	G_5	—	—	—	—	G_5 →	9C5

test for end of word

BLOCK FUNCTION: Test for last Digit (Dec.)

MAIN ROUTINE: Print Decimal

265

PC BLOCK #

Count in from PC # 265 (Test for Last Digit)

Skip in from PC # --

Explanation:

E, F, H and J recirculate.

Shift (G) to the right four bits.

Set K true for skip back to PC 261.

Count out to PC # --

Skip out to PC # 261 (Set up A_1-A_4)

BLOCK / WORD-TIME / PROPOSITION CHART

O / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0			←	—	—	—	G_1	—	—	—	—	—	—	→	7B3
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k												///			9B2

BLOCK FUNCTION: Shift (G) right

MAIN ROUTINE: Print decimal numbers

266

PC BLOCK #

Count in from PC # ----

Skip in from PC # 250 (Marker in G) or 276 [Shift (G) right]

Explanation:

See PC 215 for general type of operation, with these notes:

Set up A_1-A_3 from E using G_5, A_7 and A_8, as usual.

Set A_4-A_6 from previous contents of A_1-A_3 at $P_0\,0_0$.

Set K false for count to PC 271.

Count out to PC # 271 Shift G

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

0 / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION	
E_0	←						E_5							→		6B2
F_0	←						F_2							→		7D3
G_0	←						G_5							→		7B2
H_0	←						H_2							→		7A1
J_0	←						J_2							→		8A5
$_0k$												///				9A4
a_1	←						G_5	E_5	—					→		C7
$_0a_1$	←						G_5	E_5'	—					→		C3
a_2	←						A_7	E_5	—					→		B8
$_0a_2$	←						A_7	E_5'	—					→	12	C2
a_3	←						A_8	E_5	—					→		A8
$_0a_3$	←						A_8	E_5'	—					→		B1
a_4														A_1		A8
$_0a_4$														A_1'		A2
a_5														A_2		D2
$_0a_5$														A_2'		C3
a_6														A_3		C2
$_0a_6$														A_3'	18	B3
a_7	←						G_5	—						→		B3
$_0a_7$	←						G_5'	—						→		A2
a_8	←						A_7	—						→		B1
$_0a_8$	←						A_7'	—						→		A2

BLOCK FUNCTION: Set up A_1-A_3

MAIN ROUTINE: Print alphabetical

270

PC BLOCK #

Count in from PC # 270 (Set up A_1-A_3)

Skip in from PC # --

Explanation:

E, F, H and J recirculate.

Shift Marker in G three bits to right.

K stays false for count to PC 272.

shift 3 bits right →

Count out to PC # 272 (Set up A_1-A_6)

Skip out to PC # --

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0		←	—	—	—	—	G_2	—	—	—	—	—	—	→	7C2
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5

BLOCK FUNCTION: Shift Marker in G

MAIN ROUTINE Print alphabetical

271

PC BLOCK #

Count in from PC # 271 (Shift Marker in G)

Skip in from PC # ----

Explanation:

Set up A_1-A_6. See PC 270.

Count out to PC # 273 (Actuate Key)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	LOCATION
E_0	←							E_5							→	6B2
F_0	←							F_2	—						→	7D3
G_0	←							G_5	—						→	7B2
H_0	←							H_2	—						→	7A1
J_0	←							J_2	—						→	8A5
a_1	←							G_5	E_5	—					→	C7
${}_0a_1$	←							G_5	E_5'	—					→	C3
a_2	←							A_7	E_5	—					→	B8
${}_0a_2$	←							A_7	E_5'	—					→	12 C2
a_3	←							A_8	E_5	—					→	A8
${}_0a_3$	←							A_8	E_5'	—					→	B1
a_4															A_1	A8
${}_0a_4$															A_1'	A2
a_5															A_2	D2
${}_0a_5$															A_2'	C3
a_6															A_3	C2
${}_0a_6$															A_3'	18 B3
a_7	←							G_5	—						→	B3
${}_0a_7$	←							G_5'	—						→	A2
a_8	←							A_7	—						→	B1
${}_0a_8$	←							A_7'	—						→	A2

BLOCK FUNCTION: Set up A_1-A_6

MAIN ROUTINE Print alphabetical

272

PC BLOCK #

Count in from PC # 272

Skip in from PC # ----

Explanation:

E, F, G, H and J recirculate.

Actuate Flexowriter Key

Stick in block until GO AHEAD ends.

Count out to PC # 274

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	◄							E_5						►	6B-2
F_0	◄							F_5						►	7D-3
G_0	◄							G_5						►	7B-2
H_0	◄							H_5						►	7A-1
J_0	◄							J_5						►	8A-5
k	← Go Ahead →														
ok												///			

BLOCK FUNCTION: Actuate Flex. Key

MAIN ROUTINE: Print

273

PC BLOCK #

Count in from PC # 273

Skip in from PC # ----

Explanation:

All registers recirculate.

Stick in block until "Go Ahead" signal from Flex goes true.

Count out to PC # 275

Skip out to PC # Stick until "Go Ahead" goes true

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	
F_0	←						F_2							→	
G_0	←						G_5							→	
H_0	←						H_2							→	
J_0	←						J_2							→	
K												111			
oK	←					Go	Ahead							→	

BLOCK FUNCTION: Wait for Flex to finish printing

MAIN ROUTINE: Print

274

PC BLOCK #

Count in from PC # 274 (Wait for Flex)

Skip in from PC # ----

Explanation:

See PC 221, with these notes:

If K left false, count to PC 276.

If K is set true, skip to PC 231. Set K false at $P_2{}^0{}_{13}$ for operation in PC 231.

stypes

BLOCK / WORD-TIME / PROPOSITION CHART

	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
P	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	
E_0	←						E_5							→	6B2
F_0	←						F_2							→	7D3
G_0	←						G_5							→	7B2
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
k		→	←				F_2	G_5						G_5 →	9C5
$\circ k$	x'														9A4

end of word

Count out to PC # 276 [Shift (G) right]

Skip out to PC # 231 (Add 1 to m_1 of H)

BLOCK FUNCTION: Test Last Digit

MAIN ROUTINE: Print Alphabetical

275

PC BLOCK #

Count in from PC # 275 (Test for Last Digit)

Skip in from PC #

Explanation:

See PC 222

Shift (G) to the right three bits.

K set true for skip to PC 270.

Shift G right 3 bit

Count out to PC # ----

Skip out to PC # 270 (Set up A_1 - A_3)

BLOCK / WORD-TIME / PROPOSITION CHART

O / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0			←	—	—	—	G_2	—	—	—	—	—	—	→	7C2
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k												///			9B2

BLOCK FUNCTION: Shift (G) right

MAIN ROUTINE: Print Alphabetical

276

PC BLOCK #

Count in from PC # ----

Skip in from PC # 16 (Command Split)

Explanation:

E, F, G and J recirculate.

Test in each case for next command.

Add 1 to I (the Instruction section) of H in each word-time using K as carry FF.

When I becomes 20, K remains true.

PC No.		Description	Skip to PC
400	17	Test switch or Tape search	61 (Set up Control Number to H)
401	16	Read Tape	601 (Set up TDM Number FF's)
402	15	Write on Tape	642 (Set up TDM Number FF's)
403	14	Block search	405 (Set up FF's)

Count out to PC # Next block in series (PC 403→ PC 404)

Skip out to PC # See Explanation

BLOCK / WORD-TIME / PROPOSITION CHART

O / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_o	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D-3
G_o	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2
H_o	[	①←	—	—	—	—	H_2	—	—	—	—	—	—	→	7A3
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k												111			9B2
$_o k$	[	H_2' ② →													9A5
				①	H_o =	$(H_2$	K' +	H_2'	$K)$	$(O_{12}$ +	P_o	$O_{13})$			unit adder
				②	$_oK$ =	H_2'	$(O_{12}$ +	P_o	$O_{13})$						

BLOCK FUNCTION: Test for Command / Command Identification

MAIN ROUTINE:

400-403

PC BLOCK #

Count in from PC # 403 Test for (Block search)

Skip in from PC # ----

Explanation:

E, F, G and J recirculate.

Return control number, m_2 of G, to H.

Set K true for skip to PC 410.

Count out to PC # ----

Skip out to PC # 410 (Test for po)

BLOCK / WORD-TIME / PROPOSITION CHART

O	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT
P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	LOCATION
E_o	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_o	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_o	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2
H_o	←	—	H_2	—	—	→	←	G_5	—	→	←	—	H_2	→	7A4
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k												///			9B2

BLOCK FUNCTION: Return Control Number to H

MAIN ROUTINE: Command Identification

404

PC BLOCK #

Count in from PC # ----

Skip in from PC # 403 Test for Block search

Explanation:

E, F, G, H and J recirculate.

A_1-A_{11} and L_1-L_5 are set up according to the TDM number and block number in H.

A_9-A_{11} carry the TDM number.

A_1-A_6 have the 6 LSBD.

L_1-L_5 contain the next 5 binary digits.

A_7-A_8 have the last 2 BD's of Tape Block desired (MSBD).

K is set false for count to PC 406 and then is set true at $P_2 0_{13}$ leaving PC 405.

① $l_1 = H_2 P_o 0_2$; $_o l_1 = H'_2 P_o 0_2$

② $l_2 = H_2 P_1 0_2$; $_o l_2 = H'_2 P_1 0_2$

③ $l_3 = H_2 P_2 0_2$; $_o l_3 = H'_2 P_2 0_2$

④ $l_4 = H_2 P_o 0_3$; $_o l_4 = H'_2 P_o 0_3$

⑤ $l_5 = H_2 P_1 0_3$; $_o l_5 = H'_2 P_1 0_3$

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION	
E_o	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2	
F_o	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3	
G_o	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2	
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1	
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5	
k	1														9C5	
$_o k$												III			9A4	
a_1														H_2		C6
a_2														H_2	12	C6
a_3														H_2		B6
a_4													H_2			A7
a_5													H_2			D3
a_6												H_2				C2
a_7											H_2					B2
a_8										H_2					18	A1, B1
a_9								H_2								A1, B1
a_{10}								H_2								A1
a_{11}								H_2								A1
l_1 $_o l_1$												①				D1-2
l_2 $_o l_2$												②				D1, C2
l_3 $_o l_3$												③			10	D1, B2
l_4 $_o l_4$											④					B1-2
l_5 $_o l_5$											⑤					B1, A2

Count out to PC # 406 (Wait TDM)

Skip out to PC # ----

BLOCK FUNCTION: Set up flip-flops with tape address

MAIN ROUTINE Block search of tape

405

PC BLOCK #

Count in from PC # 405 (Set up FF's)

Skip in from PC # ----

Explanation:

E, F, G, H, J recirculate.

K, which was set true by $P_2 0_{13}$ in PC 405 and thus enters this block true, remains true and causes sticking in PC 406 until reset by the TDM set up signal.

Count out to PC # 407 (Shift Control Number to H)

Skip out to PC # Stick until TDM is set up.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
$_0K$	←	—	—	—	—	TDM		SETUP		—	—	—	—	→	9A3

BLOCK FUNCTION: Wait for relay (TDM) signal

MAIN ROUTINE Block search - tape

406

PC BLOCK #

Count in from PC # 406 (Wait for TDM signal)

Skip in from PC # ----

Explanation:

E, F and J recirculate.

Place control number in m_2 of H.

Set A_7 false. (To prevent an erroneous overflow halt print out.)

Set K true for skip to PC 6.

Count out to PC # ----

Skip out to PC # 6 (Select channel of m_2 of H)

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	7B2
F_o	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
H_o	←	—	—	H_2	—	→	←	—	G_5	→	←	—	H_2	→	7A4
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k												///			9B2
$_oa_7$														///	18A3

BLOCK FUNCTION: Dummy: Control Number to H

MAIN ROUTINE Block search

407

PC BLOCK #

Count in from PC # ----

Skip in from PC # 404 (Return Control Number to H)

Explanation:

See PC 400 - 403

PC No.	Description	Skip to PC
410	Punch Octal (IBM)	520 (Start IBM)
411	Punch Decimal (IBM)	500 (Start IBM)
412	Fill	426 (Set up Flex)
413	Extra	434 (Set up to print Control No.)
414	Extra	434
415	Read IBM Cards	455 (Start IBM)
416	Load Buffer	626 (Select channel of m_3)
417	Read Out of Buffer	436 (Select channel of m_3)

Count out to PC # Next Block in series (PC 417→PC 420)

Skip out to PC # (See Explanation)

BLOCK / WORD-TIME / PROPOSITION CHART

O / P	13 2	12 1	11 2	10 1	9 2	8 1	7 2	6 1	5 2	4 1	3 2	2 1	1 1	0 2	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2
H_0		① ←	—	—	—	—	—	—	H_2	—	—	—	—	→	7A3
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k												111			9B2
$_0k$		H_2' ②													9A5
					①	$H_0 =$	$(H_2 K' +$		$H_2' K)$	$(O_{12} + P_0 O_{13})$					
					②	$_0k =$	$H_2' ($	$O_{12} +$	$P_0 O_{13})$						

BLOCK FUNCTION: Test for next command
Command Identification

MAIN ROUTINE

410-417

PC BLOCK #

Count in from PC # 417 (Start of Buffer Out)

Skip in from PC # ----

Explanation:

G, H, J recirculate.

Set K true for skip to PC 434 indicating command error.

Count out to PC # ----

Skip out to PC # 434 (Set up to Print Control Number)

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
G_0	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
K												///			9B2

BLOCK FUNCTION: Error

MAIN ROUTINE End of Test for Command

420

PC BLOCK #

Count in from PC # ----

Skip in from PC # 412 (Test for fill from paper tape)

Explanation:

H and J recirculate.

Place ones in 0_{0-1} of G_0 to count 64 word times in next block.

Set K false for count to PC 427.

Count out to PC # 427 (Wait for RCR.)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
G_0													///	///	7B2
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
$_0K$												///			9A4

BLOCK FUNCTION: Set up to read paper tape

MAIN ROUTINE: Read paper tape

426

PC BLOCK #

Count in from PC # 426 (Set up for paper tape fill)

Skip in from PC # ----

Explanation:

H and J recirculate.

Subtract 1 from markers previously set in m_3 of G_5 during each word time.

This results in sticking in this block for 64 word times, because K follows G_5 and is set true each time until G_5 is cleared. Then K is left false and PC counts to 430.

Note: Relay Driver B-3 is energized during block 427, which actuates RCR (reader control relay) in Flexowriter.

Count out to PC # 430 (Dummy)

Skip out to PC # Stick for 64 word-times

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
G_0			←	—	—	G_5	$K+G_5'K'$	—	—	—	—	—	—	→	7C4
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k			←	—	—	—	G_5	—	—	—	—	—	—	→	9C2
σk	X'														9A4

must subt.

BLOCK FUNCTION: Wait for Flex Reader Control Relay

MAIN ROUTINE: Read paper tape

427

PC BLOCK #

Count in from PC # 427 (Wait for Flex)

Skip in from PC # ----

Explanation:

H and J recirculate.

Set K true for skip to PC 0 since K must enter false from PC 427.

Count out to PC # ----

Skip out to PC # 0 (Rest)

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	COMPONENT LOCATION
H_0		←						H_2							→	7A1
J_0		←						J_2							→	8A5
k													III			9B2

BLOCK FUNCTION: Dummy
Read paper tape

MAIN ROUTINE

430

PC BLOCK #

Count in from PC # ----

Skip in from PC # 10,413 or 414 (Extra) or 420 (Error)

Explanation:

H, J recirculate.

Put (G) into E and clear G.

Put special marker in $P_2 0_{12}$ of F to indicate error print.

Count out to PC # ----

Skip out to PC # 214 (Marker in G -- print routine)

BLOCK / WORD-TIME / PROPOSITION CHART

O / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						G_5							→	6B3
F_0		1													7D3
H_0	←			H_2										→	7A2
J_0	←						J_2							→	8A5

BLOCK FUNCTION: Set up to print Control Number

MAIN ROUTINE: Print out if no command

434

PC BLOCK #

Count in from PC # ----

Skip in from PC # 417 (Test for "Buffer Out")

Explanation:

H and J recirculate.

Set L_1-L_5 to select the channel of m_3 of H_2.

Set K false for count to PC 437.

Count out to PC # 437 (Find sector of m_2 of H)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	
H_0		←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0		←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
$_0K$													111			9A4
l_1													H_2			D1-2
$_0l_1$													H_2'			D1-2
l_2													H_2			D1, C2
$_0l_2$													H_2'			D1, C2
l_3													H_2			10 D1, B2
$_0l_3$													H_2'			D1, B2
l_4												H_2				B1-2
$_0l_4$												H_2'				B1-2
l_5												H_2				B1, A2
$_0l_5$												H_2'				B1, A2

BLOCK FUNCTION: Select channel of m_3 of H
Record (Buffer) into Memory

MAIN ROUTINE:

436

PC BLOCK #

Count in from PC # 436 (Select channel of m_3)

Skip in from PC # ----

Explanation:

H and J recirculate.

See typical sector look-up explanation, PC 7.

When M_w and H digits coincide, K is left false for count to PC 440. Set R_2 true at the same time according to equation (3).

Note: In the "Buffer Out" operation, m_3 of H will ordinarily be a main memory address. Thus L_5 is false and only equation (1) and (3) will apply.

Count out to PC # 440 ((Buffer) into Memory)

Skip out to PC # Stick until sector is found

BLOCK / WORD-TIME / PROPOSITION CHART

	0 P	13 2	12 1	11 0	10 2	9 1	8 0	7 2	6 1	5 0	4 2	3 1	2 0	1 2	0 1	0 0	COMPONENT LOCATION
H_0		←						H_2								→	7A1
J_0		←						J_2								→	8A5
k														←(1)		→(2)	9D5
σk		X'															9A4
Y_2		(3)															19B1

(1) $k = (M_w H'_2 + M'_w H_2) L'_5 O_{1-2}$

(2) $k = (M_w H'_2 + M'_w H_2) L_5 L'_4 O_1$

(3) $Y_2 = K' L'_5 P_2 O_{13}$

BLOCK FUNCTION: Find sector of m_3 of H

MAIN ROUTINE: Record (Buffer) into memory

437

PC BLOCK #

Count in from PC # 437 (Find sector of m_3)

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

In these seven blocks (PC 440-446) seven of the eight words in J are transferred into the memory since flip-flop R_2 is on. The eighth word is transferred during PC 447.

K stays false for count from block to block.

True record = $E_5 R_1 + J_2 R_2$

False record = $E'_5 R_1 + J'_2 R_2$

Count out to PC # 447 [Finish (J) into M]

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0		←						E_5							→	6B2
F_0		←						F_2							→	7D3
H_0		←						H_2							→	7A1
J_0		←						J_2							→	8A5
R_2		①														
						①	R_2	$=(K' L'_5)$		$P_1 P_{13}$						

BLOCK FUNCTION: Record (J) into memory

MAIN ROUTINE: Record (buffer) into memory

440-446

PC BLOCK #

Count in from PC # 446 [(J) into M]

Skip in from PC # ----

Explanation:

Same as 440-446, but also set K true for skip to PC 6 and set R_2 false at $P_1 0_{13}$.

Count out to PC # ----

Skip out to PC # 6 (Find channel of m_2 of H)

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_o	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k												///			9B2
$_o r_2$	!														19A1

BLOCK FUNCTION: Finish (J) into M
Record (buffer) into memory

MAIN ROUTINE:

447

PC BLOCK #

Count in from PC # ----

Skip in from PC # 415 (Test for read IBM Card)

Explanation:

Start IBM by operating start relay. Wait for One-Shot No. 1 to turn K false for count to PC 456.

Set markers in 0_0 of G for 8-word delay required in PC 456.

Count out to PC # 456 (8-word delay)

Skip out to PC # Stick until One-Shot No. 1 goes on.

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0														///	7B3
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
$_0k$	←	—	—	—	—	One	Shot	No.	1	—	—	—	—	→	9A4

BLOCK FUNCTION: Start IBM

MAIN ROUTINE Read IBM Card

455

PC BLOCK #

Count in from PC # 455 (Start IBM)

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Use K to subtract 1 from (G) in each word-time for 8 word-times.

(Note markers put in G in PC 455).

K remains false at end of 8 word-times (G) = 0 for count to PC 457.

Count out to PC # 457 (Test on R Row)

Skip out to PC # Stick for 8 word-times.

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_o	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_o			←	—	—	G_5K	+	$G_5'K'$	—	—	—	—	—	→	7C4
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_o	←	—	—	—	—	—	J_2	-	—	—	—	—	—	→	8A5
k			←	—	—	—	G_5	—	—	—	—	—	—	→	9C2
$_o k$	X'														9A4

BLOCK FUNCTION: 8-word Delay

MAIN ROUTINE Read IBM Card

456

PC BLOCK #

Count in from PC # 456 (8-word delay)

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Set K true if octal card for skip to PC 557; K remains false if decimal card for count to PC 460.

Octal is indicated by a 12-punch in the column connected to the #20 counter-total-exit on the plug-board.

Count out to PC # 460 (Dummy)

Skip out to PC # 557 (Dummy)

BLOCK / WORD-TIME / PROPOSITION CHART

0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT
P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	LOCATION
E_0	←						E_5							→	6B2
F_0	←						F_2							→	7D3
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
k	←						B_{d-0}							→	9B2

BLOCK FUNCTION: Test on R Row

MAIN ROUTINE: Read IBM

457

PC BLOCK #

Count in from PC # 457 (Test on R Row)

Skip in from PC # ----

Explanation:

K stays false for count to PC 461, then is set true at $P_2 0_{13}$ for sticking in PC 461.

Count out to PC # 461 (Wait for IBM)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
K	X														9C5

BLOCK FUNCTION: Dummy

MAIN ROUTINE Read IBM card (decimal)

460

PC BLOCK #

Count in from PC # 460 (Dummy)

Skip in from PC # 466 (Test for End)

Explanation:

E, F, H and J recirculate.

K stays true for sticking in PC 461 until one-shot No. 1 goes on and sets K false for count to PC 462. One-Shot #1 indicates next row ("11" row first time) is being read.

Count out to PC # 462 [Add (F) to (E)]

Skip out to PC # Stick until One-Shot #1 goes on

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
$_0K$	←	—	—	—	One Shot No. 1									→	9A4

BLOCK FUNCTION: Wait for IBM One-Shot #1

MAIN ROUTINE Read IBM Card

461

PC BLOCK #

Count in from PC # 461 (Wait for IBM)

Skip in from PC # ----

Explanation:

F, H and J recirculate.

Add (F) to (E) and place the sum in E.

Set A_1 true, A_2-A_4 false.

Fill markers into G_0 for an 8-word delay.

Set channel selector to 20 (L_5 true, L_4 false).

E contains in each decimal (or octal) digit the pulse pattern that a hole in a column in each row will represent. F contains the increment by which each digit in E is to be increased for each row.

K is carry flip-flop for addition, then is set false at $P_1 0_{13}$ for count.

Count out to PC # 463 (8-word delay)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	
E_0		←	E_5	F_2	$K\,A$	E_5	F_2'	K'	+	E_5'	F_2	$K'+$	E_5'	$F_2'K$	→	6A4
F_0		←						F_2							→	7D3
G_0															///	7B3
H_0		←						H_2							→	7A1
J_0		←						J_2							→	8A5
k			←				E_5	F_2							→	9B5
$\circ k$			←				E_5'	F_2'							→	9A5
a_1															///	D6
$\circ a_2$															///	C3
$\circ a_3$															///	B2
$\circ a_4$															///	A3
$\circ l_4$															///	B1-2
l_5															///	B1, A2

12

10

① = E_2 P_2 0_{13}

BLOCK FUNCTION: Add (F) to (E)

MAIN ROUTINE Read IBM Card

462

PC BLOCK #

Count in from PC # 462 [Add (F) to (E)]

Skip in from PC # ----

Explanation:

See explanation PC 456.

At end of 8 word-times, count to next PC.

Count out to PC # 464 (Wait for Buffer)

Skip out to PC # Stick for 8 word-times.

BLOCK / WORD-TIME / PROPOSITION CHART

0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT
P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0			←	—	—	$G_5 K + G_5' K'$	—	—	—	—	—	—	—	→	7C4
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
ck			←	—	—	—	G_5	—	—	—	—	—	—	→	9C2
$_0ck$	X'														9A4

BLOCK FUNCTION: Eight-word delay

MAIN ROUTINE: Read IBM card

463

PC BLOCK #

Count in from PC # 463 (8-word delay)

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Wait for Buffer register, until 2000 or 2004 is found. (H holds ø or 4 in LSOD to indicate into which half of buffer card data will be filled).

Put markers in $P_{0-1}\,0_0$ of G for four-word delay in next block.

Count out to PC # 465 (Fill Buffer)

Skip out to PC # Stick until word of buffer designated by m_3 of H is found

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	
E_0		←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0		←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0															11	7B1
H_0		←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0		←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k														← ①	② →	9D5
$_0k$		x'														9A4
							①	$k=$	$(M_4$	$H_2'+M_4'$	$H_2)$	$L_5'\,0_{0-1}$				
							②	$k=$	$(M_4$	$H_2'+M_4'$	$H_2)$	$L_5 L_4'\,0_0$				

BLOCK FUNCTION: Wait for Buffer

MAIN ROUTINE: Read IBM cards

464

PC BLOCK #

Count in from PC # 464 (Wait for Buffer)

Skip in from PC # ----

Explanation:

E, F and H recirculate.
Fill buffer register from IBM Card. Stick in PC 465 for 4 words, counting in G (subtracting from markers put in 0_0 of G in PC 464, using K).

Step A_1, A_2, A_3, A_4 at end of each word. Data is read into J as follows.

Card columns are divided into four groups of 10 columns each. One group is read each word-time. During a word and while the IBM is at each row, each column of the group is examined. If it has a punch,(E)is transferred to J during that decimal digit time. If not, J recirculates.

Example: For "2" row.

E =00 0010 0010 0010 0010 0010 0010 0010 0010 0010 0010 0010

Suppose columns 1, 3, 4, 5 and 9 of this group of 10 are punched in this row---number 2 x 222 xxx 2 x---Then we have

J =00 0010 0000 0010 0010 0010 0000 0000 0000 0010 0000

These are the equations for J_0:

$1 = E_5 (B_{1-2}A_1 + B_{2-2}A_2 + B_{3-2}A_3 + B_{4-2}A_4) + J_2 (B_{1-2}A_1 + B_{2-2}A_2 + B_{3-2}A_3 + B_{4-2}A_4)'$

$2 = E_5 (B_{1-3}A_1 + B_{2-3}A_2 + B_{3-3}A_3 + B_{4-3}A_4) + J_2 (B_{1-3}A_1 + B_{2-3}A_2 + B_{3-3}A_3 + B_{4-3}A_4)'$

$3 = E_5 (B_{1-4}A_1 + B_{2-4}A_2 + B_{3-4}A_3 + B_{4-4}A_4) + J_2 (B_{1-4}A_1 + B_{2-4}A_2 + B_{3-4}A_3 + B_{4-4}A_4)'$

$4 = E_5 (B_{1-5}A_1 + B_{2-5}A_2 + B_{3-5}A_3 + B_{4-5}A_4) + J_2 (B_{1-5}A_1 + B_{2-5}A_2 + B_{3-5}A_3 + B_{4-5}A_4)'$

$5 = E_5 (B_{1-6}A_1 + B_{2-6}A_2 + B_{3-6}A_3 + B_{4-6}A_4) + J_2 (B_{1-6}A_1 + B_{2-6}A_2 + B_{3-6}A_3 + B_{4-6}A_4)'$

$6 = E_5 (B_{1-7}A_1 + B_{2-7}A_2 + B_{3-7}A_3 + B_{4-7}A_4) + J_2 (B_{1-7}A_1 + B_{2-7}A_2 + B_{3-7}A_3 + B_{4-7}A_4)'$

$7 = E_5 (B_{1-8}A_1 + B_{2-8}A_2 + B_{3-8}A_3 + B_{4-8}A_4) + J_2 (B_{1-8}A_1 + B_{2-8}A_2 + B_{3-8}A_3 + B_{4-8}A_4)'$

$8 = E_5 (B_{1-9}A_1 + B_{2-9}A_2 + B_{3-9}A_3 + B_{4-9}A_4) + J_2 (B_{1-9}A_1 + B_{2-9}A_2 + B_{3-9}A_3 + B_{4-9}A4)'$

$9 = E_5(B_{1-10}A_1 + B_{2-10}A_2 + B_{3-10}A_3 + B_{4-10}A_4) + J_2(B_{1-10}A_1 B_{2-10}A_2 + B_{3-10}A_3 + B_{4-10}A_4)'$

$10 = E_5(B_{1-11}A_1 + B_{2-11}A_2 + B_{3-11}A_3 + B_{4-11}A_4) + J_2(B_{1-11}A_1 B_{2-11}A_2 + B_{3-11}A_3 + B_{4-11}A_4)'$

Count out to PC # 466 (Test for End)

Skip out to PC # Stick for four word-times

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	
E_0		←							E_5						→	6B2
F_0		←							F_2						→	7D3
G_0				←			G_5	K	+	G_5'	K'				→	7C4
H_0									H_2						→	7A1
J_0		←	①	②	③		④	⑤		⑥	⑦	⑧		⑨	⑩ →	8A3 (etc.)
k				←					G_5						→	9C2
$_0k$		X'														9A4
$_0a_1$		X'														D2
a_2		A_1														B6
$_0a_2$		A_1'														B3
a_3		A_2														12 A8
$_0a_3$		A_2'														A4
a_4		A_3														A8
$_0a_4$		A_3'														A4

BLOCK FUNCTION: Fill Buffer (4 words)

MAIN ROUTINE Read IBM Card (decimal)

465

PC BLOCK #

Count in from PC # 465 (Fill Buffer)

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Equation (1) goes true for any configuration in $0_0 + P_0 0_1$ other than 9 (1001), in which case K is set true and the PC skips to 461 to read another row. When (E) is all 9's, K remains false.

Count out to PC # 467 (Dummy)

Skip out to PC # 461 (Wait for IBM)

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B2
F_0	←						F_2							→	7D3
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
k													←	(1) →	9C4
		(1)	k=	E_5'	$P_0 0_1$	+ E_5	$P_2 0_0$	+ E_5	$P_1 0_0$	+ E_5'	$P_0 0_0$				

BLOCK FUNCTION: Test for End of Card

MAIN ROUTINE Read IBM card (decimal)

466

PC BLOCK #

Count in from PC # 466 (Test for End)

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Set K true.

Count out to PC # ----

Skip out to PC # 6 (Select channel of m_9 of H)

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B2
F_0	←						F_2							→	7D3
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
k												///			9B2

BLOCK FUNCTION: Dummy

MAIN ROUTINE Read IBM card

467

PC BLOCK #

Count in from PC # ----

Skip in from PC # 411 (Test for pd)

Explanation:

E, F, H and J recirculate.

Start IBM, wait for One Shot No. 2 which sets K false for count to next PC.

Set L_1-L_5 for buffer (channel 20).

Put marker in 0_0 of G for 8-word delay in next block.

Count out to PC # 501 (8-word delay)

Skip out to PC # Stick until One-Shot No. 2 is on

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_o	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_o														///	7B3
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
$_oK$	←	—	—	One	Shot		No.2	—	—	—	—	—	—	→	9A4
$_oL_4$														///	B1-2
L_5														///	10 B1, A2

BLOCK FUNCTION: Start IBM

MAIN ROUTINE: Punch decimal card

500

PC BLOCK #

Count in from PC # 500 (Start IBM)

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Use K to subtract 1 from (G) in each word-time for 8 word-times.

(Note markers put in G in PC 455).

K remains false at end of 8 word-times (G) =0 for count to PC 457.

Count out to PC # 502 (Wait for Buffer)

Skip out to PC # Stick for 8 word-times

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0			←	—	G_5	K	+	G_5'	K'	—	—	—	—	→	7C4
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
ℓ			←	—	—	—	G_5	—	—	—	—	—	—	→	9C2
$_0\ell$	X'														9A4

BLOCK FUNCTION: 8-word delay

MAIN ROUTINE: Pd

501

PC BLOCK #

Count in from PC # 501 (8-word delay)

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Wait for Buffer register, until 2000 or 2004 is found. (H holds ∅ or 4 in LSOD to indicate into which half of buffer card data will be filled).

Put markers in P_{0-1} 0_0 of G for four-word delay in next block.

Count out to PC # 503

Skip out to PC # Stick until buffer ready

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0														//	7B1
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k													← ①	→ ②	9D5
$_0k$	X'														9A4
				①	$k = (M_w\ H_2' + M_w'\ H_2)\ L_5'\ O_{1-2}$										
				②	$k = (M_w\ H_2' + M_w'\ \frac{H}{2})\ L_5\ L_4'\ O_1$										

BLOCK FUNCTION: Wait for Buffer

MAIN ROUTINE: Punch card or Read Card (563)

502

PC BLOCK #

Count in from PC # 502 (Wait for Buffer)

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Wait for 4 words, while K subtracts 1 each time from markers previously placed in 0_0 of G.

Then K goes false for count to PC 504. (or to PC 530, like PC 510).

Count out to PC # 504 (Wait for IBM)

Skip out to PC # Stick for four word-times.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0			←	—	—	G_5K	$+G_5'$	K'	—	—	—	—	—	→	7C4
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k			←	—	—	—	G_5	—	—	—	—	—	—	→	9C2
$_0k$	X'														9C4

BLOCK FUNCTION: 4-word delay

MAIN ROUTINE : Punch decimal card

503

PC BLOCK #

Count in from PC # 503 (4-word delay)

Skip in from PC # ----

Explanation:

E, F, H, J recirculate.

Set markers in 0_0 of G for 8-word delay.

IBM One-Shot No. 2 sets K false for count to next PC block.

Count out to PC # 505 (8-word delay)

Skip out to PC # Stick until IBM One-Shot No. 2 on.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_o	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_o														///	7B3
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
$\not{k}$												///			9B2
$_o\not{k}$	←	—	—	—	One Shot No. 2									→	9A4

BLOCK FUNCTION: Wait for IBM

MAIN ROUTINE: Punch decimal card

504

PC BLOCK #

Count in from PC # 504

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Use K to subtract 1 from (G) in each word-time for 8 word-times.

(Note markers put in G in PC 455).

K remains false at end of 8 word-times (G) = 0 for count to PC 457.

Count out to PC # 506

Skip out to PC # Stick for 8 word-times.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0			←	—	—	G_5K'	+	$G_5'K'$						→	7C4
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k			←	—	—	—	G_5	—	—	—	—	—	—	→	9C2
ok	X'														9A4

BLOCK FUNCTION: 8-word Delay

MAIN ROUTINE : pd

505

PC BLOCK #

Count in from PC # 505

Skip in from PC # -----

Explanation:

E, F, H and J recirculate.

Wait for Buffer register, until 2000 or 2004 is found. (H holds ø or 4 in LSOD to indicate into which half of buffer card data will be filled).

Put markers in P_{0-1} 0_0 of G for four-word delay in next block.

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B2
F_0	←						F_2							→	7D3
G_0														11	7B1
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
K													←①	→②	9D5
$\sim K$	X'														9A4
						①	$K=(M_\omega H_2' + M_\omega' H_2) L_5' 0_{4-1}$								
						②	$K=(M_\omega H_2' + M_\omega' H_2) L_5 L_4' 0_0$								

Count out to PC # 507

Skip out to PC # Stick until word of buffer designated by m_3 of H is found

BLOCK FUNCTION: Wait for Buffer

MAIN ROUTINE: pd

506

PC BLOCK #

Count in from PC # 506 (Wait for Buffer)

Skip in from PC # ----

Explanation:

Set up thyratrons by equations on drawing 20213014.

Repeat the block 4 words (until (G) = 0) with A_1 true first word, A_2 true second word, A_3 third word, A_4 fourth word-time.

Thyratrons are set up to punch first group of 10 columns during first word; 2nd group during 2nd word, etc. They are set to punch in any row and column for which (E) and (J) agree during the decimal digit that represents that row and column.

A_5 is true at 1st pulse of following decimal digit when (J) and (E) agree during entire decimal digit.

A_6 remembers for one decimal digit that A_5 indicated agreement during a decimal digit (4 bits). If A_6 is true, the thyratron that allows the card to be punched is fired.

Count out to PC # 510 (Dummy)

Skip out to PC # Stick for 4 words

BLOCK / WORD-TIME / PROPOSITION CHART

0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0			←	—	G_5	K	+	G_5'	K'	—	—	—	—	→	7C4
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k			←	—	—	—	G_5	—	—	—	—	—	—	→	9C2
$_0k$	X'														9A4
$_0a_1$	X'														D2
a_2	A_1														B6
$_0a_2$	A_1'														B3
a_3	A_2														A8 (12)
$_0a_3$	A_2'														A4
a_4	A_3														A8
$_0a_4$	A_3'														A4
a_5		①		①	①	①		①	①	①		①	①	①	D2
$_0a_5$	←	—	—	—	—	J_2	E_5'	+	J_2'	E_5	—	—	—	→	C2 (18)
a_6	A_5	A_5		A_5	A_5	A_5		A_5	A_5	A_5		A_5	A_5	A_5	C2
$_0a_6$	1		1	1	1		1	1	1		1	1	1		B2
	①	$a_5 = (J_2$	E_5 +	J_2'	E_5'	$[P_0(O_{0,4,8,12}) + P_1(O_{1,5,9}) + P_2(O_{2,6,10})]$									

BLOCK FUNCTION: 4-word Delay. Set up X-row thyratrons

MAIN ROUTINE: Punch card (decimal)

507

PC BLOCK #

Count in from PC # 507

Skip in from PC # ----

Explanation:

At end of block, with K remaining false for count to next block, set K to enter next block true.

Count out to PC # 511 (Wait for IBM)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 2	12 1 0	11 2	10 1 0	9 2	8 1 0	7 2	6 1 0	5 2	4 1 0	3 2	2 1 0	1 2	0 1 0	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k	X														9C5

BLOCK FUNCTION: Dummy

MAIN ROUTINE: Punch decimal card

510

PC BLOCK #

Count in from PC # 510 (Dummy)

Skip in from PC # 516 (Test for End)

Explanation:

E, F, H and J recirculate.

Wait for one-shot #2. Next row ready to punch -- zero row first time through.

K enters true and is set false when one-shot #2 goes on.

Count out to PC # 512 (Add F to E)

Skip out to PC # Stick until One-Shot #2 goes on.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
$_0K$	←	—	—	—	One Shot #2									→	9A4

BLOCK FUNCTION: Wait for IBM

MAIN ROUTINE: Punch card

511

PC BLOCK #

Count in from PC # 511 (Wait for IBM)

Skip in from PC # ----

Explanation:

Same as PC 462, except L_4 and L_5 are not set up.

Count out to PC # 513 (8-word delay)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	
E_0		①	←	E_3	F_2	$K+$	E_5	F_2'	$K'+$	$E'F_2$	K'	$+E_5'$	F_2'	K'	→	6A4
F_0		←						F_2							→	7D3
G_0															///	7B3
H_0		←						H_2							→	7A1
J_0		←						J_2							→	8A5
k			←				E_5	F_2							→	9B5
$_0k$		1	←				E_5'	F_2'							→	9A5
a_1															///	D6
$_0a_2$															///	C3
$_0a_3$															///	B2
$_0a_4$															///	A3
$_0a_5$															///	C3
$_0a_6$															///	B3

① = E_2 P_2 O_{13}

BLOCK FUNCTION: Add (F) to (E)

MAIN ROUTINE Punch card, dec.

512

PC BLOCK #

Count in from PC # 512

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Use K to subtract 1 from (G) in each word-time for 8 word-times.

(Note markers put in G in PC 455).

K remains false at end of 8 word-times (G) =0 for count to PC 457.

Count out to PC # 514

Skip out to PC # Stick for 8 word-times

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B2
F_0	←						F_2							→	7D3
G_0			←			G_5K	+	$G_5'K'$						→	7C4
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
k			←				G_5							→	9C2
ok	X'														9A4

BLOCK FUNCTION: 8-word Delay

MAIN ROUTINE: pd

513

PC BLOCK #

Count in from PC # 513

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Wait for Buffer register, until 2000 or 2004 is found. (H holds ø or 4 in LSOD to indicate into which half of buffer card data will be filled).

Put markers in P_{0-1} 0_0 of G for four-word delay in next block.

Count out to PC # 515

Skip out to PC # Stick until buffer ready

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_o	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_o														//	7B1
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
K													← ①	→ ②	9D5
$_oUK$	X'														9A4
						①	$K = (M\omega H_2' + M'\omega H_2) L_5' O_{q-1}$								
						②	$K = (M\omega H_2' + M'\omega H_2) L_5 4' O_0$								

BLOCK FUNCTION: Wait for Buffer

MAIN ROUTINE: pd

514

PC BLOCK #

Count in from PC # 514

Skip in from PC # ----

Explanation:

Set up thyratrons by equations on drawing 20213014.

Repeat the block 4 words (until (G) =0) with A_1 true first word, A_2 true second word, A_3 third word, A_4 fourth word-time.

Thyratrons are set up to punch first group of 10 columns during first word; 2nd group during 2nd word, etc. They are set to punch in any row and column for which (E) and (J) agree during the decimal digit that represents that row and column.

A_5 is true at 1st pulse of following decimal digit when (J) and (E) agree during entire decimal digit.

A_6 remembers for one decimal digit that A_5 indicated agreement during a decimal digit (4 bits). If A_6 is true, the thyratron that allows the card to be punched is fired.

Count out to PC # 510 (Dummy)

Skip out to PC # Stick for 4 words

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B2
F_0	←						F_2							→	7D3
G_0			←		G_5	K	+	G_5'	K'					→	7C4
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
k			←				G_5							→	9C2
$_0k$	X'														9A4
a_{01}	X'														D2
a_2	A_1														B6
a_{02}	A_1'														B3
a_3	A_2														12 A8
a_{03}	A_2'														A4
a_4	A_3														A8
$_0a_4$	A_3'														A4
a_5		①		①	①	①		①	①	①		①	①	①	D2
$_0a_5$	←					J_2	E_5'	+	J_2'	E_5				→	18 C2
a_6	A_5	A_5		A_5	A_5	A_5		A_5	A_5	A_5		A_5	A_5	A_5	C2
$_0a_6$	1		1	1	1		1	1	1		1	1	1		B2
	①	$a_5=(J_2$	E_5+	J_2'	$E_5')$										

BLOCK FUNCTION: Set up digit thyratrons

MAIN ROUTINE : pd

515

PC BLOCK #

Count in from PC # 515 (Set thyration - 4 word-times). (See PC 507).

Skip in from PC # ----

Explanation:

Test for last row.

Set up last thyratron, with A_5-A_6 making comparison between (E) and (J).

When E is all nines (1001), k will fail to true; set K so PC will count to PC 517. Until the 9-row is punched, K will be set true for skip back to PC 511 to repeat punching operation.

BLOCK / WORD-TIME / PROPOSITION CHART

O / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_o	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k													←	① →	9C4
a_6	A_5	A_5		A_5	A_5	A_5		A_5	A_5	A_5		A_5	A_5	A_5	18C2
$_0a_6$	1		1	1	1		1	1	1		1	1	1		18B2
	①	$k = E_5' P_0 Q_1 + E_5 P_2 Q_0 + E_5 P_1 Q_0 + E_5' P_0 Q_0$													

Count out to PC # 517 (Dummy)

Skip out to PC # 511 (Wait for IBM)

BLOCK FUNCTION: Test for End

MAIN ROUTINE Punch card (d.)

516

PC BLOCK #

Count in from PC # 516 (Test for End)

Skip in from PC # ----

Explanation:

E, F, H, J recirculate.

K is set true for skip to PC 6.

Count out to PC # ----

Skip out to PC # 6 (Select channel of m_2 of H)

BLOCK / WORD-TIME / PROPOSITION CHART

O	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT
P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k												///			9B2

BLOCK FUNCTION: Dummy

MAIN ROUTINE: Punch Card decimal

517

PC BLOCK #

Count in from PC # ----

Skip in from PC # 410 (Test for p o)

Explanation:

E, F, H and J recirculate.

Start IBM, wait for One Shot No. 2 which sets K false for count to next PC.

Set L_1 -L_5 for buffer (channel 20).

Put marker in 0_0 of G for 8-word delay in next block.

Count out to PC # 521 (8-word delay)

Skip out to PC # Stick until One-Shot No. 2 is on

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0														///	7B3
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
$_0k$	←	—	—	One Shot	—	—	No.2	—	—	—	—	—	—	→	9A4
$_0l_4$														///	B1-2
l_5														///	B1, A2

10

BLOCK FUNCTION: Start IBM

MAIN ROUTINE: Punch Octal card

520

PC BLOCK #

Count in from PC # 520

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Use K to subtract 1 from (G) in each word-time for 8 word-times.

(Note markers put in G in PC 455).

K remains false at end of 8 word-times (G) =0 for count to PC 457.

Count out to PC # 522

Skip out to PC # Stick for 8 word-times.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←						E_5							→	6B2
F_o	←						F_2							→	7D3
G_o			←			G_5K	+	$G_5'K'$						→	7C4
H_o	←						H_2							→	7A1
J_o	←						J_2							→	8A5
k			←				G_5							→	9C2
.k	X'														9A4

BLOCK FUNCTION: 8-word Delay

MAIN ROUTINE: po

521

PC BLOCK #

Count in from PC # 521

Skip in from PC # -----

Explanation:

E, F, H and J recirculate.

Wait for Buffer register, until 2000 or 2004 is found. (H holds ∅ or 4 LSOD to indicate into which half of buffer card data will be filled).

Put markers in P_{0-1} 0_0 of G for four-word delay in next block.

Count out to PC # 523

Skip out to PC # Stick until buffer ready

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B2
F_0	←						F_2							→	7D3
G_0														11	7B1
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
K													←①	②→	9D5
$.K$	x'														9A4
						①	$K=(M\omega\ H_2'+M'\omega\ H_2)\ L_5'\ 0_{0-1}$								
						②	$K=(M\omega\ H_2'+M'\omega\ H_2)\ L_5\ L_4'\ 0_0$								

BLOCK FUNCTION: Wait for Buffer

MAIN ROUTINE: po

522

PC BLOCK #

Count in from PC # 522 (Wait for Buffer. See PC 502).

Skip in from PC # ----

Explanation:

See PC 503.

(See PC 504, similar to PC 524; then PC 501, similar to PC 525;

PC 502 like PC 526; PC 503 like PC 527; PC 510 like PC 530; PC 511 like PC 531;

PC 512 like PC 532; PC 501 like PC 533; PC 502 like PC 534; count to PC 535).

Thyratron punches 12 in column 20.

Count out to PC # 524 (Wait for IBM)

Skip out to PC # Stick for four words.

BLOCK / WORD-TIME / PROPOSITION CHART

0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0			←	—	G_5	K	+	G_5'	K'	—	—	—	—	→	7C4
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k			←	—	—	—	G_5	—						→	9C2
ok	X'														9C4

BLOCK FUNCTION: Set up R Row Thyration.

MAIN ROUTINE Punch Card (o.)

523

PC BLOCK #

Count in from PC # 523 (Set up R Row Thyratron)

Skip in from PC # ----

Explanation:

E, F, H, J recirculate.

Set markers in 0_0 of G for 8-word delay.

IBM One-Shot No. 2 sets K false for count to next PC block.

Count out to PC # 525

Skip out to PC # Stick until IBM One-Shot No. 2 on.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_o	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_o														///	7B3
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
K												///			9B2
$_o K$	←	—	—	—	One Shot No. 2									→	9A4

BLOCK FUNCTION: Wait for IBM

MAIN ROUTINE: po

524

PC BLOCK #

Count in from PC # 524

Skip in from PC # -----

Explanation:

E, F, H and J recirculate.

Use K to subtract 1 from (G) in each word-time for 8 word-times.

(Note markers put in G in PC 455).

K remains false at end of 8 word-times (G) = 0 for count to PC 457.

Count out to PC # 527

Skip out to PC # Stick for 8 word-times.

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5							→	6B2
F_0	←	—	—	—	—	—	F_2							→	7D3
G_0			←	—	—	G_5K	+	G'_5K'						→	7C4
H_0	←	—	—	—	—	—	H_2							→	7A1
J_0	←	—	—	—	—	—	J_2							→	8A5
k			←	—	—	—	G_5							→	9C2
₀k	X'														9A4

BLOCK FUNCTION: 8-word Delay

MAIN ROUTINE: po

525

PC BLOCK #

Count in from PC # 525

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Wait for Buffer register, until 2000 or 2004 is found. (H holds $\emptyset$ or 4 in LSOD to indicate into which half of buffer card data will be filled).

Put markers in P_{0-1} 0_0 of G for four-word delay in next block.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B2
F_0	←						F_2							→	7D3
G_0														11	7B1
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
k													←①	→②	9D5
ok	x'														9A4
						①	k=	$(M_w$	$H_2'+$	M_w'	$H_2)$	L_5'	0_{q-1}		
						②	k=	$(M_w$	$H_2'+$	M_w'	$H_2)$	L_5L_4'	0_0		

Count out to PC # 527

Skip out to PC # Stick until word of buffer designated by m_3 of H is found.

BLOCK FUNCTION: Wait for Buffer

MAIN ROUTINE: po

526

PC BLOCK #

Count in from PC # 526

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Wait for 4 words, while K subtracts 1 each time from markers previously placed in 0_0 of G.

Then K goes false for count to PC 504. (Or to PC 530, like PC 510).

Count out to PC # 530

Skip out to PC # Stick for four word-times.

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B2
F_0	←						F_2							→	7D3
G_0			←			G_5K	$+G'_5K'$							→	7C4
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
k			←				G_5							→	9C2
$_0k$	X'														9C4

BLOCK FUNCTION: 4-word delay

MAIN ROUTINE : po

527

PC BLOCK #

Count in from PC # 527

Skip in from PC # ----

Explanation:

At end of block, with K remaining false for count to next block, set K to enter next block true.

Count out to PC # 531

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B2
F_0	←						F_2							→	7D3
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
k	X														9C5

BLOCK FUNCTION: Dummy

MAIN ROUTINE: po

530

PC BLOCK #

Count in from PC # 530

Skip in from PC # 536

Explanation:

E, F, H and J recirculate.

Wait for one-shot #2. Next row ready to punch -- zero row first time through.

K enters true and is set false when one-shot #2 goes on.

Count out to PC # 532

Skip out to PC # Stick until One-Shot #2 goes on.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0 ←							E_5							→	6B2
F_0 ←							F_2							→	7D3
H_0 ←							H_2							→	7A1
J_0 ←							J_2							→	8A5
ok ←					One Shot #2									→	9A4

BLOCK FUNCTION: Wait for IBM

MAIN ROUTINE: po

531

PC BLOCK #

Count in from PC # 531

Skip in from PC # ----

Explanation:

Same as PC 462, except L_4 and L_5 are not set up.

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	① ←	E_5	F_2	$K+$	F_5	F'_2	K'	+	E'_5	F_2	$K'+$	E'_5	F'_2K	→	6A4
F_0	←						F_2							→	7D3
G_0														III	7B3
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
k		←				E_5	F_2							→	9B5
$_0k$	1 ←					E_5	F'_2							→	9A5
a_1														III	D6
$_0a_2$														III	C3
$_0a_3$														III	B2
$_0a_4$														III	A3

12

① = $E_2 \bar{P}_2 O_{13}$

Count out to PC # 533

Skip out to PC # ----

BLOCK FUNCTION: Add (F) to (E)

MAIN ROUTINE: po

532

PC BLOCK #

Count in from PC # 532

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Use K to subtract 1 from (G) in each word-time for 8 word-times.

(Note markers put in G in PC 455).

K remains false at end of 8 word-times (G) = 0 for count to PC 457.

Count out to PC # 534

Skip out to PC # Stick for 8 word-times.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0			←	—	—	G_5K	+	G'_5K'						→	7C4
H_0	←	—	—	—	—	—	H_2							→	7A1
J_0	←	—	—	—	—	—	J_2							→	8A5
k			←	—	—	—	G_5							→	9C2
ok	X'														9A4

BLOCK FUNCTION: 8-word Delay

MAIN ROUTINE: po

533

PC BLOCK #

Count in from PC # 533

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Wait for Buffer register, until 2000 or 2004 is found. (H holds $\emptyset$ or 4 in LSOD to indicate into which half of buffer card data will be filled).

Put markers in P_{0-1} 0_0 of G for four-word delay in next block.

Count out to PC # 535

Skip out to PC # Stick until word of buffer designated by m_3 of H is found

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0														//	7B1
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k													← ①	② →	9D5
ok	X'														9A4
						①	k =	$(M_\omega$	H_2' +	M_ω'	$H_2)$	L_5'	O_{e-1}		
						②	k =	$(M_\omega$	H_2' +	M_ω'	$H_2)$	L_5	$L_4' O_0$		

BLOCK FUNCTION: Wait for Buffer

MAIN ROUTINE: po

534

PC BLOCK #

Count in from PC # 534 (Wait for Buffer) (See PC 502)

Skip in from PC # ----

Explanation:

Similar to PC's 507 and 515.

Again A_5 is true when(J)and(E)agree. A_6 is set true if A_5 indicates agreement after each octal digit (3 bits). If A_6 is true the thyratron that allows the card to be punched is set up. For balance of explanation see PC 507.

Count out to PC # 536 (Test for End)

Skip out to PC # Stick four word-times.

BLOCK / WORD-TIME / PROPOSITION CHART

	0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0		←						E_5							→	6B2
F_0		←						F_2							→	7D3
G_0			←				G_5K +		$G_5'K'$						→	7C4
H_0		←						H_2							→	7A1
J_0		←						J_2							→	8A5
k			←					G_5							→	9C2
$_0k$		X'														9A4
$_0a_1$		X'														D2
a_2		A_1														B6
$_0a_2$		A_1'														B3
a_3		A_2														A8 (12)
$_0a_3$		A_2'														A4
a_4		A_3														A8
$_0a_4$		A_3'														A4
a_5		①	①	①	①	①	①	①	①	①	①	①	①	①	①	D2
$_0a_5$		←					J_2E_5' +		$J_2'E_5$						→	C2 (18)
a_6		④	④	④	④	④	④	④	④	④	④	④	④	④	④	C2
$_0a_6$		⑤	⑤	⑤	⑤	⑤	⑤	⑤	⑤	⑤	⑤	⑤	⑤	⑤	⑤	B2
			①	$a_5 = (J_2E_5 + J_2'E_5')P_0O_{0-3}$												
			④	$a_6 = [A_5(J_2E_5 + J_2'E_5')[P_2$												
			⑤	$_0a_6 = [A_5(J_2E_5 + J_2'E_5')[P_2$												

BLOCK FUNCTION: Set Thyratrons (4 words)

MAIN ROUTINE Punch card (octal)

535

PC BLOCK #

Count in from PC # 535 (Set Thyrations)

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Last thyration set up, with A_5-A_6 making comparison between (E) and (J).

When (E) is all sevens (111), K will remain false for count to PC 537. Until the 7-row is punched, K will be set true, causing skip back to PC 531.

Count out to PC # 537 (Dummy)

Skip out to PC # 531 (Wait for IBM)

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←						E_5							→	6B2
F_o	←						F_2							→	7D3
H_o	←						H_2							→	7A1
J_o	←						J_2							→	8A5
k			←					E's						→	9B5
a_6	①	①	①	①	①	①	①	①	①	①	①	①	①	①	18C2
$a_o a_6$	②	②	②	②	②	②	②	②	②	②	②	②	②	②	18B2

① $A_5 (J_2 E_5 + J_2' E_2')\, P_2$

② $[A_5 (J_2 E_5 + J_2' E_2')]' P_2$

BLOCK FUNCTION: Test for End

MAIN ROUTINE: Punch card (o.)

536

PC BLOCK #

Count in from PC # 536

Skip in from PC # ----

Explanation:

E, F, H, J recirculate.

K is set true for skip to PC 6.

Count out to PC # ----

Skip out to PC # 6 (Select channel of m_3 of H)

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	P	210	210	210	210	210	210	210	210	210	210	210	210	210	210	
E_0	←							E_5							→	6B2
F_0	←							F_2							→	7D3
H_0	←							H_2							→	7A1
J_0	←							J_2							→	8A5
k													111			9B2

BLOCK FUNCTION: Dummy

MAIN ROUTINE: po

537

PC BLOCK #

Count in from PC # ----

Skip in from PC # 457 (Test on R Row)

Explanation:

E, F, H, J recirculate.

Set K false for count to PC 560, then in $P_2 0_{13}$ set K true for next block (PC 560).

Count out to PC # 560 (Wait for IBM)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_1	—	—	—	—	—	—	→	7D3
H_0	←	—	—	—	—	—	H_1	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_1	—	—	—	—	—	—	→	8A5
ck	X														9C5
$_0$ck												///			9A4

BLOCK FUNCTION: Dummy

MAIN ROUTINE Read IBM card (o.)

557

PC BLOCK #

Count in from PC # 557

Skip in from PC # 565

Explanation:

E, F, H and J recirculate.

K stays true for sticking in PC 461 until one-shot No. 1 goes on and sets K false for count to PC 462. One-Shot #1 indicates next row ("11" row first time) is being read.

Count out to PC # 561 (Add (F) to (E)

Skip out to PC # Stick until One-Shot #1 goes on

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B2
F_0	←						F_2							→	7D3
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
$_0k$	←				One Shot No. 1									→	9A4

BLOCK FUNCTION: Wait for IBM one-shot #1

MAIN ROUTINE: Read IBM card

560

PC BLOCK #

Count in from PC # 560

Skip in from PC # ----

Explanation:

F, H and J recirculate.

Add (F) to (E) and place the sum in E.

Set A_1 true, A_2-A_4 false.

Fill markers into G_0 for an 8-word delay.

Set channel selector to 20 (L_5 true, L_4 false).

E contains in each decimal (or octal) digit the pulse pattern that a hole in a column in each row will represent. F contains the increment by which each digit in E is to be increased for each row.

K is carry flip-flop for addition, then is set false at $P_1 0_{13}$ for count.

Count out to PC # 562

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	⓪←	E_5	F_2	K_+	E_5	F_2'	K'	+	E_5'	F_2	K_+'	E_5'	F_2'	K→	6A4
F_0	←						F_2							→	7D3
G_0														///	7B3
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
K		←				E_5	F_2							→	9B5
$_0K$	1	←				E_5'	F_2'							→	9A5
a_1														///	D6
$_0a_2$														///	C3
$_0a_3$														///	B2
$_0a_4$														///	A3
$_0l_4$														///	B1-2
l_5														///	B1, A2
			⓪=	E_2	P_2	0_{13}									

12

10

BLOCK FUNCTION: Add (F) to (E)

MAIN ROUTINE: Read IBM card

561

PC BLOCK #

Count in from PC # 561

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Use K to subtract 1 from (G) in each word-time for 8 word-times.

(Note markers put in G in PC 455).

K remains false at end of 8 word-times (G) =0 for count to PC 457.

Count out to PC # 563

Skip out to PC # Stick for 8 word-times.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B2
F_0	←						F_2							→	7D3
G_0			←			G_5K	+	$G_5'K'$						→	7C4
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
k			←				G_5							→	9C2
ok	X'														9A4

BLOCK FUNCTION: 8-word Delay

MAIN ROUTINE: Read IBM card

562

PC BLOCK #

Count in from PC # 562

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Wait for Buffer register, until 2000 or 2004 is found. (H holds $\emptyset$ or 4 in LSOD to indicate into which half of buffer card data will be filled).

Put markers in P_{0-1} 0_0 of G for four-word delay in next block.

Count out to PC # 564

Skip out to PC # Stick until word of buffer designated by m_3 of H is found.

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0														11	7B1
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k													←①	②→	9D5
$_0$k	X'														9A4
						①	k =	$(M_\omega H'_2 + M'_\omega$			$H_2)$	L'_5	0_{a-1}		
						②	k =	$(M_\omega H'_2 + M'_\omega$			$H_2)$	L'_5	$L'_4 0_0$		

BLOCK FUNCTION: Wait for Buffer

MAIN ROUTINE: Read IBM card

563

PC BLOCK #

Count in from PC # 563 (Wait for Buffer) See PC 464.

Skip in from PC # ----

Explanation:

Similar to PC 465, except that card columns are divided into four groups of 14 columns each, with one group read each word time. Otherwise, the same explanation applies.

E 00 010 010 - - - -)
J 0 0 010 000 - - - -) in example similar to that of PC 465

These are the equations for J_0:

$1 = E_5 (B_{1-1}A_1 + B_{2-1}A_2 + B_{3-1}A_3 + B_{4-1}A_4) + J_2 (B_{1-1}A_1 + B_{2-1}A_2 + B_{3-1}A_3 + B_{4-1}A_4)'$

$2 = E_5 (B_{1-2}A_1 + B_{2-2}A_2 + B_{3-2}A_3 + B_{4-2}A_4) + J_2 (B_{1-2}A_1 + B_{2-2}A_2 + B_{3-2}A_3 + B_{4-2}A_4)'$

$3 = E_5 (B_{1-3}A_1 + B_{2-3}A_2 + B_{3-3}A_3 + B_{4-3}A_4) + J_2 (B_{1-3}A_1 + B_{2-3}A_2 + B_{3-3}A_3 + B_{4-3}A_4)'$

$4 = E_5 (B_{1-4}A_1 + B_{2-4}A_2 + B_{3-4}A_3 + B_{4-4}A_4) + J_2 (B_{1-4}A_1 + B_{2-4}A_2 + B_{3-4}A_3 + B_{4-4}A_4)'$

$5 = E_5 (B_{1-5}A_1 + B_{2-5}A_2 + B_{3-5}A_3 + B_{4-5}A_4) + J_2 (B_{1-5}A_1 + B_{2-5}A_2 + B_{3-5}A_3 + B_{4-5}A_4)'$

$6 = E_5 (B_{1-6}A_1 + B_{2-6}A_2 + B_{3-6}A_3 + B_{4-6}A_4) + J_2 (B_{1-6}A_1 + B_{2-6}A_2 + B_{3-6}A_3 + B_{4-6}A_4)'$

$7 = E_5 (B_{1-7}A_1 + B_{2-7}A_2 + B_{3-7}A_3 + B_{4-7}A_4) + J_2 (B_{1-7}A_1 + B_{2-7}A_2 + B_{3-7}A_3 + B_{4-7}A_4)'$

$8 = E_5 (B_{1-8}A_1 + B_{2-8}A_2 + B_{3-8}A_3 + B_{4-8}A_4) + J_2 (B_{1-8}A_1 + B_{2-8}A_2 + B_{3-8}A_3 + B_{4-8}A_4)'$

$9 = E_5 (B_{1-9}A_1 + B_{2-9}A_2 + B_{3-9}A_3 + B_{4-9}A_4) + J_2 (B_{1-9}A_1 + B_{2-9}A_2 + B_{3-9}A_3 + B_{4-9}A_4)'$

$10 = E_5(B_{1-10}A_1 + B_{2-10}A_2 + B_{3-10}A_3 + B_{4-10}A_4) + J_2(B_{1-10}A_1 + B_{2-10}A_2 + B_{3-10}A_3 + B_{4-10}A_4)'$

$11 = E_5(B_{1-11}A_1 + B_{2-11}A_2 + B_{3-11}A_3 + B_{4-11}A_4) + J_2(B_{1-11}A_1 + B_{2-11}A_2 + B_{3-11}A_3 + B_{4-11}A_4)'$

$12 = E_5(B_{1-12}A_1 + B_{2-12}A_2 + B_{3-12}A_3 + B_{4-12}A_4) + J_2(B_{1-12}A_1 + B_{2-12}A_2 + B_{3-12}A_3 + B_{4-12}A_4)'$

$13 = E_5(B_{1-13}A_1 + B_{2-13}A_2 + B_{3-13}A_3 + B_{4-13}A_4) + J_2(B_{1-13}A_1 + B_{2-13}A_2 + B_{3-13}A_3 + B_{4-13}A_4)'$

$14 = E_5(B_{1-14}A_1 + B_{2-14}A_2 + B_{3-14}A_3 + B_{4-14}A_4) + J_2(B_{1-14}A_1 + B_{2-14}A_2 + B_{3-14}A_3 + B_{4-14}A_4)'$

Count out to PC # 565 (Test for End)

Skip out to PC # Stick for 4 words

BLOCK / WORD-TIME / PROPOSITION CHART

	0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0		←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0		←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0				←	—	—	—	$G_5K + G_5'$	K'	—	—	—	—	—	→	7C4
H_0		←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0		①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩	⑪	⑫	⑬	⑭	8A4 (etc.)
K				←	—	—	—	G_5	—	—	—	—	—	—	→	9C2
$_0k$		X'														9A4
$_0a_1$		X'														D2
a_2		A_1														B6
$_0a_2$		A_1'														B3
a_3		A_2														12< A8
$_0a_3$		A_2'														A4
a_4		A_3														A8
$_0a_4$		A_3'														A4

BLOCK FUNCTION: Fill Buffer (4 word)

MAIN ROUTINE Read card (o.)

564

PC BLOCK #

Count in from PC # 564 (Fill Buffer - 4 word)

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Test for end of IBM card.

When (E) is all sevens (111), K will remain false permitting count to PC 566.

Until all 7's are read, K is set true for skip back to PC 560 to continue reading card.

Count out to PC # 566 (Dummy)

Skip out to PC # 560 (Wait for IBM)

BLOCK / WORD-TIME / PROPOSITION CHART

O / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B2
F_0	←						F_2							→	7D3
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
k			←											E_5' →	9B5

BLOCK FUNCTION: Test for End.

MAIN ROUTINE: Read IBM Card (o.)

565

PC BLOCK #

Count in from PC # 565

Skip in from PC # ----

Explanation:

E, F, H and J recirculate.

Set K true.

Count out to PC # ----

Skip out to PC # 6

BLOCK / WORD-TIME / PROPOSITION CHART

	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	P	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	COMPONENT LOCATION
E_o	◄	—	—	—	—	—	—	E_5	—	—	—	—	—	—	►	6B2
F_o	◄	—	—	—	—	—	—	F_2	—	—	—	—	—	—	►	7D3
H_o	◄	—	—	—	—	—	—	H_2	—	—	—	—	—	—	►	7A1
J_o	◄	—	—	—	—	—	—	J_2	—	—	—	—	—	—	►	8A5
K													///			9B2

BLOCK FUNCTION: Dummy

MAIN ROUTINE: Read IBM card

566

PC BLOCK #

Count in from PC # ----

Skip in from PC # 401 (Test for read tape)

Explanation:

Start tape and actuate read relay.

H and J recirculate.

Copy m_3 of H into G. (Address of first cell in the buffer to be read into.)

Set up A_7 -A_8 to "mode".

Set A_9-A_{11} to tape unit number (TDM#), held in H_2.

K stays true for skip to PC 605.

Set channel selector to buffer.

$A'_7A'_8$: read address word (w_0) from tape

$A_7A'_8$: read 8 odd words (w_1, w_3, ---w_{15})

A'_7A_8: read 8 even words (w_2, w_4, ---w_{16})

Count out to PC # ----

Skip out to PC # 605 (Wait for First C_t)

BLOCK / WORD-TIME / PROPOSITION CHART

O / P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
G_0							←	G_5		→			H_1 →		7B4
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
a_7										H_2					18B2
a_8										H_2					18A-B1
a_9								H_2							A1
a_{10}								H_2							18 A1
a_{11}								H_2							A1
ek_4														///	
l_5														///	

BLOCK FUNCTION: Set up TDM # Flip-Flops

MAIN ROUTINE Read Tape

601

PC BLOCK #

Count in from PC # ----

Skip in from PC # 601 (Set up TDM# FF's) or 623 (Test for End)

Explanation:

G, H and J recirculate.

K remains true until there is a tape clock pulse.

When a C_t is received, four bits of information from the tape (T_0-T_3) are read into L_1-L_4 in parallel and K is set false for count to PC 606. K is reset true at $P_2 0_{13}$.

In next word time, A_5 and K are to be true.

Count out to PC # 606 (Second tape clock C_t)

Skip out to PC # Stick until tape clock pulse is received

BLOCK / WORD-TIME / PROPOSITION CHART

	0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
G_0		←						G_5							→	7B2
H_0		←						H_2							→	7A1
J_0		←						J_2							→	8A5
k		K'														9C5
$_0k$		←						A_{13}							→	9A2
a_5		K'														18D1
$_0a_5$		①														18C3
l_1		←						T_0	C_t						→	D4
$_0l_1$		←						T_0'	Ct						→	D4
l_2		←						T_1	Ct						→	C4
$_0l_2$		←						T_1'	Ct						→	C4
l_3		←						T_2	C_t						→	10 C4
$_0l_3$		←						T_2'	Ct						→	B4
l_4		←						T_3	C_t						→	B4
$_0l_4$		←						T_3'	C_t						→	A4
			①	$_0a_5$	A_5	K	$P_2 0_{13}$									
a_{13}		←							C_t						→	
$_0a_{13}$		←							K'						→	

BLOCK FUNCTION: Wait for First Tape Clock Pulse

MAIN ROUTINE Read Tape

605

PC BLOCK #

Count in from PC # 650 (Wait for First C_t)

Skip in from PC # ----

Explanation:

G, H and J recirculate.

During the first time through this block, A_5 is true. This transfers the information in L_1-L_4 into A_1-A_4 at $P_0 0_0$.

(E) is shifted right four bits and (A_1-A_4) is shifted into E during 0_{12} + $P_0 0_{13}$ of this first word-time. Note that E_0 follows E_1 to perform shift, except when it follows A_1 to get new information. Information from tape comes LSD first.

A_5 is reset to zero unless another tape clock has appeared. K entered block true causing PC to stick in the block until the C_t pulse comes along. When the C_t comes, information is transferred into L_1-L_4 and operation proceeds as in PC 605.

The same operations repeat in PC's 606-617 through 11 clock pulses. This results in shifting in 44 bits of information, with the last four bits (of which only 1 is useful) in L_1-L_4. (The word is 42 bits long, but the 42nd bit is always zero).

The following are the grid equations for the synchronizing flip-flop A_{13}.

$$a_{13} = C_t$$

$${}_0a_{13} = K'C$$

Count out to PC # 620 (Fill $P_1 0_{13}$ into E. Wait for word in J)

Skip out to PC # Stick until C_t is received

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
		①	$E_0 = (E_5 A'_5 + A_1 A_5)(0_{12} + P_0 0_{13})$												
E_0		①					$E_5 A'_5 + E_1 A_5$							→	6C4, 5
G_0	←						G_5							→	7B2
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
k	K'														9C5
${}_0k$	←						A_{13}							→	9A2
a_1		A_2 →												L_1	12C6
${}_0a_1$		A'_2 →												L'_1	12C2
a_2		A_3 →												L_2	12B7
${}_0a_2$		A'_3 →												L'_2	12B3
a_3		A_4 →												L_3	12A6
${}_0a_3$		A'_4 →												L'_3	12A3
a_4														L_4	12A8
${}_0a_4$														L'_4	12A3
a_5	K'														18D1
${}_0a_5$	A_5K														18C3
l_1	←						T_0	C_t						→	D4
${}_0l_1$	←						T'_0	C_t						→	D4
l_2	←						T_1	C_t						→	C4
${}_0l_2$	←						T'_1	C_t						→	C4
l_3	←						T_2	C_t						→	C4
${}_0l_3$	←						T'_2	C_t						→	B4
l_4	←						T_3	C_t						→	B4
${}_0l_4$	←						T'_3	C_t						→	A4

10

BLOCK FUNCTION: Wait for Next C_t (2nd through 11th)

MAIN ROUTINE Read Tape

606, 607, 610-617

PC BLOCK #

Count in from PC # 617 (11th C_t)

Skip in from PC # ----

Explanation:

G, H and J recirculate.

Data in L_1 is transferred into $P_1 0_{13}$ of E_1 completing the reading of one word (41 bits) since only L_1 now has information of interest.

L_4 is set false, L_5 set true to cause look-up equation ③ to be used.

BLOCK / WORD-TIME / PROPOSITION CHART

	0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0		①	←	—	—	—	—	E_5	—	—	—	—	—	—	→	6C3
G_0		←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2
H_0		←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0		←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k														②	③→	9D5
$_0k$		1														9A4
$_0l_4$															///	10B1
l_5															///	10B1, A2
				①			$E_0 =$	$L_1 P_1$	0_{13}							
				②	$k =$	$(M_\omega$	$H_2' +$	M_ω'	$H_2)$	L_5'	0_{0-1}					
				③	$k_0 =$	$(M_\omega$	$H_2' +$	M_ω'	$H_2)$	L_5	L_4'	0_0				

Count out to PC # 621 [Fill (E) into J]

Skip out to PC # Stick until Buffer word appears

BLOCK FUNCTION: Fill $P_1 0_{13}$ of E. Wait for Buffer

MAIN ROUTINE Read Tape

620

PC BLOCK #

Count in from PC # 620 (Fill $P_1 0_{13}$ of E. Wait for Buffer)

Skip in from PC # ---

Explanation:

G and H recirculate.

(E) is transferred into the buffer J.

Set K true, leaving the block, for adding 1 to m_3 of H in PC 622.

K stays false until $P_2 0_{13}$ for count to PC 622.

Count out to PC # 622 (Add 1 to m_2 of H)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
G_0	←						G_5							→	7B2
H_0	←						H_2							→	7A1
J_0	←						E_5							→	8A5
k	1														9C5

BLOCK FUNCTION: Fill (E) into J

MAIN ROUTINE: Read Tape

621

PC BLOCK #

Count in from PC # 621 [Fill (E) into J]

Skip in from PC # ----

Explanation:

G and J recirculate.

K was set true leaving PC 621. Hence 1 is added to m_3 of H using K for the carry flip-flop. This will allow the next word from the tape to be read into the next word in the J-line (buffer).

K is left false by ${}_0k = H'_2$ for count to PC 623.

Count out to PC # 623 (Test for End)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
P	2	1	0	2	1	0	2	1	0	2	1	0	2	1	0	2	1	0	
G_0	←						G_5							→	7B2
H_0	←			H_2								$KH'_2+K'H_2$		→	7A2
J_0	←						J_2							→	8A5
${}_0k$										←		H'_2		→	9A5

BLOCK FUNCTION: Add 1 to m_3 of H

MAIN ROUTINE Read tape

622

PC BLOCK #

Count in from PC # 622 (Add 1 to m_3 of H)

Skip in from PC # ----

Explanation:

G, H and J recirculate.

Compare m_3 of G and H in 0_0. m_3 of G holds address of first word recorded. If we have returned to that word (that is, read 8 words from tape, adding 1 to m_3 of H each time) the routine is complete and we count to PC 624, since K is left false.

If not, K is set true for skip back to PC 605 to read next word from tape. Note that if A_7 and A_8 are both false (Mode = 0) we go out of routine after reading only one word (the block address word).

Count out to PC # 624 (Dummy, Control # to H.)

Skip out to PC # 605 (Wait for 1st C_t)

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
G_0	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k														←(1)→	9C4, 5
			$k = (A_7 + A_8)(G_5 H_2' + G_5' H_2) O_0$												

BLOCK FUNCTION: Test for end

MAIN ROUTINE: Read tape

623

PC BLOCK #

Count in from PC # 623 (Test for end of r.t.)

Skip in from PC # ----

Explanation:

Transfer the control number from m_2 of G to m_2 of H.

Set A_7 false.

Set K true for skip to PC 6.

Count out to PC # ----

Skip out to PC # 6 (Select channel of m_2 of H)

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
G_0	←						G_5							→	7B2
H_0	←			H_2			→ ←	G_5		→ ←		H_2		→	7A4
J_0	←						J_2							→	8A5
k												///			9B1
$_0a_7$														///	18A3

BLOCK FUNCTION: Dummy. Control # to H.

MAIN ROUTINE Read Tape

624

PC BLOCK #

Count in from PC # ----

Skip in from PC # Test for 416 (Buffer load command)

Explanation:

H and J recirculate.

Set L_1-L_5 to select the channel from m_3 of H.

Set K false for count to PC 627.

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION	
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1	
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5	
$_0k$												///			9A4	
l_1												H_2				D1, 2
$_0l_1$												H_2'				D1, 2
l_2												H_2				D1, C2
$_0l_2$												H_2'				D1, C2
l_3												H_2			10	D1, C2
$_0l_3$												H_2'				D1, B2
l_4											H_2					B1-2
$_0l_4$											H_2'					B1-2
l_5											H_2					B1, A2
$_0l_5$											H_2'					B1, A2

Count out to PC # 627 (Select sector of m_3 of H)

Skip out to PC # ----

BLOCK FUNCTION: Select channel of m_0 of H

MAIN ROUTINE Load buffer

626

PC BLOCK #

Count in from PC # 626 (Select channel of m_3 of H)

Skip in from PC # ----

Explanation:

H and J recirculate.

Compare M_w and H_2 to find sector of m_3 of H, using K for look-up (sticking until M_w and H_2 digits coincide).

Note: In the "buffer load" operation, m_3 of H will ordinarily be a main memory address. Thus L_5 is false, and equation ① applies.

Count out to PC # 630-636 (Fill J from M)

Skip out to PC # Stick until sector is found.

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_o	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k													←	①②	9D5
$_ok$	k'														9A4

① $k = (M_w H_2' + M_w' H_2) L_5' O_{0-1}$

② $k = (M_w H_2' + M_w' H_2) L_5 L_4' O_0$

BLOCK FUNCTION: Select sector of m_3 of H

MAIN ROUTINE Load buffer

627

PC BLOCK #

Count in from PC # 627 (Find sector of m_3 of H)

Skip in from PC # ----

Explanation:

H recirculates.

Fill J from memory.

K stays false and PC counts until 8 consecutive words are filled.

Count out to PC # 637 (Fill J from M)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
H_0	←						H_2							→	7A1
J_0	←		$ML'_5 + J_2 L_5 L'_4 L'_1 + G_5 L_5 L_4$											→	8A2

BLOCK FUNCTION: Fill J from Memory

MAIN ROUTINE Load buffer

630 - 636

PC BLOCK #

Count in from PC # 636 (Fill J from M)

Skip in from PC # ----

Explanation:

H recirculates.

Complete filling buffer from memory (8th word). Set K true for skip to PC 6.

Count out to PC # ----

Skip out to PC # 6 (Find channel of m_2 of H)

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
H_o	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_o	←	—	ML'_5+J_2		L_5L_4		$L'_4L'_1+G_5$		L_5L_4	—	—	—	—	→	8A2
k												///			9B1

BLOCK FUNCTION: Fill J from M. Set K true.

MAIN ROUTINE: Load Buffer

637

PC BLOCK #

Count in from PC # ----

Skip in from PC # 402 (Test for Command)

Explanation:

G, H, J recirculate.

H holds command as usual.

TDM# (tape unit number) is transferred from 0_6 of H to A_9-A_{11}.

Transfer mode from H to A_7-A_8.

Mode 0 -write block address word only

Mode 1 -write odd words

Mode 2 -write even words

L_5 is set true, L_4 false, to read buffer.

K remains true for skip to PC 643.

Before the first tape clock is received by the computer, it is necessary to look up the first buffer word and to fill the FF digit registers (L_1-L_4, A_1-A_4) with the first digits of that word to be recorded on the tape.

Count out to PC # ----

Skip out to PC # 643 (Find lst word)

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
G_0	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
a_7										H_2					18B2
a_8										H_2					18A-B1
a_7								H_2							A1
a_{10}								H_2							18 A1
a_{11}								H_2							A1
ol_4														///	10 B1
l_5														///	10 B1, A2

BLOCK FUNCTION: Set up TDM # FF's

MAIN ROUTINE: Write Tape

642

PC BLOCK #

Count in from PC # ----

Skip in from PC # 642 (Set up TDM # FF's)

Explanation:

First word to tape is always filled from cell 2000.

Put 1 in $P_0 0_0$ of G. This will be used in PC 663 to indicate when 8 words have been recorded.

When 0_0 of M_w =0, K stays false for count to PC 644.

K set true at end of block.

Count out to PC # 644 [Fill (J) into E]

Skip out to PC # Stick until M_w indicates sector 0

BLOCK / WORD-TIME / PROPOSITION CHART

0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
G_0	←	—	—	—	—	—	G_5	—	—	→				‖	7B1
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
k													←①	②→	9D5
$_0k$	X														9A4
			①	$k=(M_\omega H'_2 + M'_\omega H_2) L'_5 O_{0-1}$											
			②	$k=(M_\omega H'_2 + M'_\omega H_2) L_5 L'_4 O'_0$											

BLOCK FUNCTION: Find 1st word

MAIN ROUTINE: Write tape

643

PC BLOCK #

Count in from PC # 643 (Find lst word)

Skip in from PC # ----

Explanation:

G and J recirculate.

Record (J) into E.

Record a marker into $P_1 0_0$ of H.

This indicates next sector look-up in J will be for cell 2002.

K stays false for count to PC 645.

Count out to PC # 645 [Fill (J) into F]

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←	—	—	—	—	—	J_4	—	—	—	—	—	—	→	6A2
G_0	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2
H_0														1	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5

BLOCK FUNCTION: Fill (J) into E

MAIN ROUTINE: Write tape

644

PC BLOCK #

Count in from PC # 644 [Fill (J) into E]

Skip in from PC # ----

Explanation:

E, G, H, J recirculate.

Record (J) into F. This stores (2001) in F.

K stays false for count to PC 646.

Count out to PC # 646 (First digit from E to SR_i)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
OP	210	210	210	210	210	210	210	210	210	210	210	210	210	210	
E_0	←	—	—	—	—	—	E_5	—	—	—	—	—	—	→	6B2
F_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	7D3
G_0	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5

BLOCK FUNCTION: Fill (J) into F

MAIN ROUTINE: Write tape

645

PC BLOCK #

Count in from PC # 645 [Fill (J) into F]

Skip in from PC # ----

Explanation:

F, G, H, J recirculate

A_1-A_4 is an intermediate static register, SR_i.

L_1-L_4 is the recording static register, SR_R.

4 LSBD are transferred from E to A_1-A_4.

E is shifted right four bits.

K stays false for count to PC 647.

Count out to PC # 647 (SR_i-SR_R; $D_1 \longrightarrow SR_i$)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 2 1 0	12 2 1 0	11 2 1 0	10 2 1 0	9 2 1 0	8 2 1 0	7 2 1 0	6 2 1 0	5 2 1 0	4 2 1 0	3 2 0	2 2 0	1 2 1 0	0 2 1 0	COMPONENT LOCATION
E_0		←	—	—	—	—	—	E_1	—	—	—	—	—	→	6B3
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3
G_0	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1
J_0	←	—	—	—	—	—	J_2	—	—	—	—	—	—	→	8A5
a_1														E_5	C5
$_0a_1$														E'_5	C1
a_2														E_5	C6
$_0a_2$														E'_5	12 B2
a_3														E_5	A6
$_0a_3$														E'_5	A2
a_4													E_5		A7
$_0a_4$													E'_5		A3

BLOCK FUNCTION: Transfer 4 LSB to SR_i

MAIN ROUTINE Write tape

646

PC BLOCK #

Count in from PC # 646 ($D_0 \rightarrow SR_i$)

Skip in from PC # ---

Explanation:

F, G, H and J recirculate.

Transfer data from A_1-A_4 to L_1-L_4.

Transfer next 4 bits from E to A_1-A_4.

Shift E right 4 bits.

K stays false for count to PC 650, and is set true in $P_2 0_{13}$ to enter PC 650 true.

At the end of this block the computer is ready for the 1st C_t . L_1-L_4 contains 1st digit, A_1-A_4 contains 2nd digit, E the last 9 digits of first word, F the 2nd word and H is ready to look for the 3rd word.

Count out to PC # 650 (Wait for C_t)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION	
E_0								E_1	—	—	—	—	—	→	6B3	
F_0	←	—	—	—	—	—	F_2	—	—	—	—	—	—	→	7D3	
G_0	←	—	—	—	—	—	G_5	—	—	—	—	—	—	→	7B2	
H_0	←	—	—	—	—	—	H_2	—	—	—	—	—	—	→	7A1	
J_0	←	—	—	—	—	—	J_4	—	—	—	—	—	—	→	8A5	
k	X'														9C5	
a_1														E_5		C5
$_0a_1$														E_5'		C1
a_2														E_5		C6
$_0a_2$														E_5'	12	B2
a_3														E_5		A6
$_0a_3$														E_5'		A2
a_4													E_5			A7
$_0a_4$													E_5'			A3
l_1														A_1		D2
$_0l_1$														A_1'		D2
l_2														A_2		C2
$_0l_2$														A_2'	10A2	C2
l_3														A_3		B2
$_0l_3$														A_3'		B2
l_4														A_4		B2
$_0l_4$														A_4'		A2

BLOCK FUNCTION: ($SR_1 \rightarrow SR_R$; $D_1 \rightarrow SR_1$)

MAIN ROUTINE: Write tape

647

PC BLOCK #

Count in from PC # 647 ($SR_i \rightarrow SR_R$; $D_1 \rightarrow SR_i$)

Skip in from PC # 673 (Add 1 to m_3 of H)

Explanation:

F, G, H and J recirculate.

When fall of C_t occurs, transfer data from A_1-A_4 to L_1-L_4 and thence to tape.

A_5 is set true at end of each block and stays true for 1st word-time in next block unless another C_t is received. It permits next 4 bits to be transferred from E to A_1-A_4, and E to be shifted right 4 bits. This process is repeated in each block.

With fall of C_t in each block, K is set false for count to next block; and then true again at end of block.

The following are the grid equations for the synchronizing flip-flop A_{13}:

$a_{13} = C_t$

$_0a_{13} = K'C$

Count out to PC # 660 (Wait for 9th C_t, last transfer)

Skip out to PC # Stick to fall of C_t in each block

BLOCK / WORD-TIME / PROPOSITION CHART

	13	12	11	10	9	8	7	6	5	4	3	2	1	0	COMPONENT LOCATION
	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	2 1 0	
E_0	←					A_5	E_1 +	A_5'	E_5					→	6A3
F_0	←						F_2							→	7D3
G_0	←						G_5							→	7B2
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
k	K'														9C5
$_0k$	←						A_{13}							→	9A2
a_1														E_5A_5	C5
$_0a_1$														$E_5'A_5$	C1
a_2														E_5A_5	C6
$_0a_2$														$E_5'A_5$	12 B2
a_3														E_5A_5	A6
$_0a_3$														$E_5'A_5$	A2
a_4													E_5A_5		A7
$_0a_4$													$E_5'A_5$		A3
a_5	K'														18 D1
$_0a_5$	A_5K														18 C3
l_1	←						A_1	C_t						→	D3
$_0l_1$	←						A_1'	C_t						→	D3
l_2	←						A_2	C_t						→	C3
$_0l_2$	←						A_2'	C_t						→	10 C3
l_3	←						A_3	C_t						→	B3
$_0l_3$	←						A_3'	C_t						→	B3
l_4	←						A_4	C_t						→	B3
$_0l_4$	←						A_4'	C_t						→	A3

BLOCK FUNCTION: Wait for C_t. Transfer word to tape

MAIN ROUTINE Write tape

650-657, 660

PC BLOCK #

Count in from PC # 660 (Wait for 9th C_t)

Skip in from PC # ----

Explanation:

Last transfer from E→ A_1-A_4. The bit originally in $P_1 0_{13}$ of E is put in A_1, A_2-A_4 are set to zero.

C_t in this block causes last transfer of (A_1-A_4) with L_1-L_4.

$a_{13} = C_t$

$_0a_{13} = K'C$

Count out to PC # 662 (Wait for 11th C_t. Transfer (F) to E)

Skip out to PC # Stick until 10th C_t falls.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
F_0	←						F_2							→	7D3
G_0	←						G_5							→	7B2
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
k	K'														9C5
$_0k$	←						A_{13}							→	9A2
a_5	K'														18D1
$_0a_5$	A_5K														18C3
l_1	←						A_1	C_t						→	D3
$_0l_1$	←						A_1'	C_t						→	D3
l_2	←						A_2	C_t						→	C3
$_0l_2$	←						A_2'	C_t						→	10 C3
l_3	←						A_3	C_t						→	B3
$_0l_3$	←						A_3'	C_t						→	B3
l_4	←						A_4	C_t						→	B3
$_0l_4$	←						A_4'	C_t						→	A3
a_1														E_5A_5	
$_0a_1$														$E_5'A_5$	
$_0a_2$														III	
$_0a_3$														III	
$_0a_4$														III	
a_{13}	←							C_t						→	
$_0a_{13}$	←							$K'C$						→	

BLOCK FUNCTION: Wait for 10th C_t. Last transfer A→L

MAIN ROUTINE Write tape

661

PC BLOCK #

Count in from PC # 661 (Wait for 10th C_t. Last transfer A→L)

Skip in from PC # ---

Explanation:

F, G, H, J recirculate.

When 11th C_t falls, recording is complete.

Transfer (F) to E.

$A_{13} = C_t$

$_oA_{13} = K'C$

Count out to PC # 663 (D_o ⟶ SR_i. Test for End)

Skip out to PC # Stick until 11th C_t falls.

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←						F_2							→	6B2
F_o	←						F_2							→	7D3
G_o	←						G_2							→	7B2
H_o	←						H_2							→	7A1
J_o	←						J_2							→	8A5
k	K'														9C5
$_ok$	←						A_{13}							→	9A2

BLOCK FUNCTION: Wait for 11th C_t. Fill (F) into E.

MAIN ROUTINE: Write tape

662

PC BLOCK #

Count in from PC # 662 (Wait for 11th C_t. (F) to E)

Skip in from PC # ----

Explanation:

G, H, J recirculate.

Transfer first 4 bits of E into A_1-A_4 ($D_0 \rightarrow SR_i$).

Shift E right 4 bits.

Test for end of routine: skip to PC 67 if mode is zero (A_7) and (A_8) =0) or if (G) and (H) are the same in 0_0. [G has a 1 in LSOD. H is 2 first time through -- one is added to H in PC 673 so H will be back to 1 in LSOD after 8th word is recorded.]

If not end of routine, count to PC 664.

A_6 is set true unless proper buffer word is found (2002 first time through). If A_6 is still false at $P_2 0_{13}$ set A_{12} true at end of word.

Count out to PC # 664 (Wait for W_2. $SR_i \rightarrow SR_R$)

Skip out to PC # 67 (Set A false. C # → H)

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0								E_1						→	6B3
G_0	←						G_5'							→	7B2
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
$_0a_k$														←①	
a_1														E_5	C5
$_0a_1$														E_5'	C1
a_2														E_5	C6
$_0a_2$														E_5'	12 B2
a_3														E_5	A6
$_0a_3$														E_5'	A2
a_4													E_5		A7
$_0a_4$													E_5'		A3
a_6														②→	C3
$_0a_6$	1														18 B3
a_{12}	A_6'														A2
$_0a_{12}$	A_6														A2
				①	$_0K = (A_7 + A_3)(G_5 H_2' + G_5' H_2) O_0$										
				②	$a_6 = (M_4 H_2' + M_4' H_2) O_0$										

BLOCK FUNCTION: Transfer $D_0 \rightarrow SR_i$. Test for End

MAIN ROUTINE: Write tape

663

PC BLOCK #

Count in from PC # 663 ($D_0 \rightarrow SR_i$. Test for End)

Skip in from PC # ----

Explanation:

① $a_6 = (M_w H'_2 + M'_w H_2)\, O_0$

Transfer 4 bits from E to A_1-A_4. Shift (E) right 4 bits. First transfer(A_1-A_4) to L_1-L_4 to be ready to record on tape.

Transfer (J) into F if A_{12} true: that is, if proper buffer word was found in PC 663.

If not, repeat search for buffer word as in PC 663.

Count out to PC # 665 (Wait for W_2)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

O P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION	
E_o								E_1							6B3	
F_o				J_2	A_{12}	+	F_2	A'_{12}							7D3	
G_o							G_5								7B2	
H_o							H_2								7A1	
J_o							J_2								8A5	
a_1														E_5		C5
${}_o a_1$														E'_5		C1
a_2														E_5		C6
${}_o a_2$														E'_5	12	B2
a_3														E_5		A6
${}_o a_3$														E'_5		A2
a_4													E_5			A7
${}_o a_4$													E'_5			A3
a_6				①	$a_6 = (M_w H'_2 + M_w' H_2)$						O_o			①		C3
${}_o a_6$	1														18	B3
a_{12}	A'_6															A2
${}_o a_{12}$	A_6															A2
l_1														A_1		D2
${}_o l_1$														A'_1		D2
l_2														A_2		C2
${}_o l_2$														A'_2	10A2	C2
l_3														A_3		B2
${}_o l_3$														A'_3		B2
l_4														A_4		B2
${}_o l_4$														A'_4		A2

BLOCK FUNCTION: Wait for Buffer Word. $SR_i \rightarrow SR_R$

MAIN ROUTINE Write Tape

664

PC BLOCK #

Count in from PC # 664 (Wait for Buffer Word. $SR_i \rightarrow SR_R$)

Skip in from PC # ----

Explanation:

E, G, H and J recirculate.

Keep looking for correct buffer word as in PC 663.

If word is found, A_6 stays false, A_{12} goes true next word time to allow transfer.

Whenever A_{12} is true, make the transfer of word to F.

Count out to PC # 672 (Search for W_2. Set K true)

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

0 P	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B2
F_0	←					$\overline{J_2} A_{12}$ +		$F_2 A'_{12}$						→	7D3
G_0	←						G_5							→	7B2
H_0	←						H_2							→	7A1
J_0	←						J_2							→	8A5
a_6														0	C3
$_0a_6$	1														18 B3
a_{12}	A'_6														A2
$_0a_{12}$	A_6														A2
			①	$a_6 =$	$(M_6 H'_2$	+	$M'_6 H_2)$	O_0							

BLOCK FUNCTION: Search for Buffer Word; Set A_6 true.

MAIN ROUTINE: Write Tape

665-667, 670-671

PC BLOCK #

Count in from PC # 671

Skip in from PC # ----

Explanation:

E, G, H and J recirculate.

K is set true at $P_2 0_{13}$ for operation in 673.

If word is found, A_6 stays false, A_{12} goes true next word-time to allow transfer.

Whenever A_{12} is true, make the transfer of word to F.

Count out to PC # 673

Skip out to PC # ----

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_0	←						E_5							→	6B2
F_0	←				J_2	A_{12}	+	F_2	A'_{12}	—				→	7D3
G_0	←						G_5	—						→	7B2
H_0	←						H_2	—						→	7A1
J_0	←						J_2							→	8A5
k	X'														9C5
a_6														①	C3
$_0a_6$	1														B3
a_{12}	A'_6														18 A2
$_0a_{12}$	A_6														A2
			①	$a_6 =$	$(M_{00}$	H'_2	$+M'_{00}$	$H_2)$	O_0						

BLOCK FUNCTION: Search for W_2. Set K true.

MAIN ROUTINE Write tape

672

PC BLOCK #

Count in from PC # 672

Skip in from PC # ----

Explanation:

Make last test for next word as in preceding blocks. Add 1 to m_3 of H for finding next word. Set K true to skip back to 650.

Count out to PC # ----

Skip out to PC # 650

BLOCK / WORD-TIME / PROPOSITION CHART

OP	13 210	12 210	11 210	10 210	9 210	8 210	7 210	6 210	5 210	4 210	3 210	2 210	1 210	0 210	COMPONENT LOCATION
E_o	←							E_5						→	
F_o	←					J_2	A_{12}+	F_2	A'_{12}					→	6B2
G_o	←							G_5						→	7D3
H_o	←			H_2						→	K	H'_2+	K'H_2	→	7B2
J_o	←							J_2						→	7A2
k	111														8A5
$_ok$											←	H'_2		→	9B2
$_oa_5$													111		9A5
a_6													①		
$_oa_6$	1														C3
a_{12}	A'_6														B3
$_oa_{12}$	A_6														A2
															A2
			① = $(M_\omega H'_2 + M'_\omega H_2)O_o$												

BLOCK FUNCTION: Add 1 to m_3 of H

MAIN ROUTINE: wt

673

PC BLOCK #

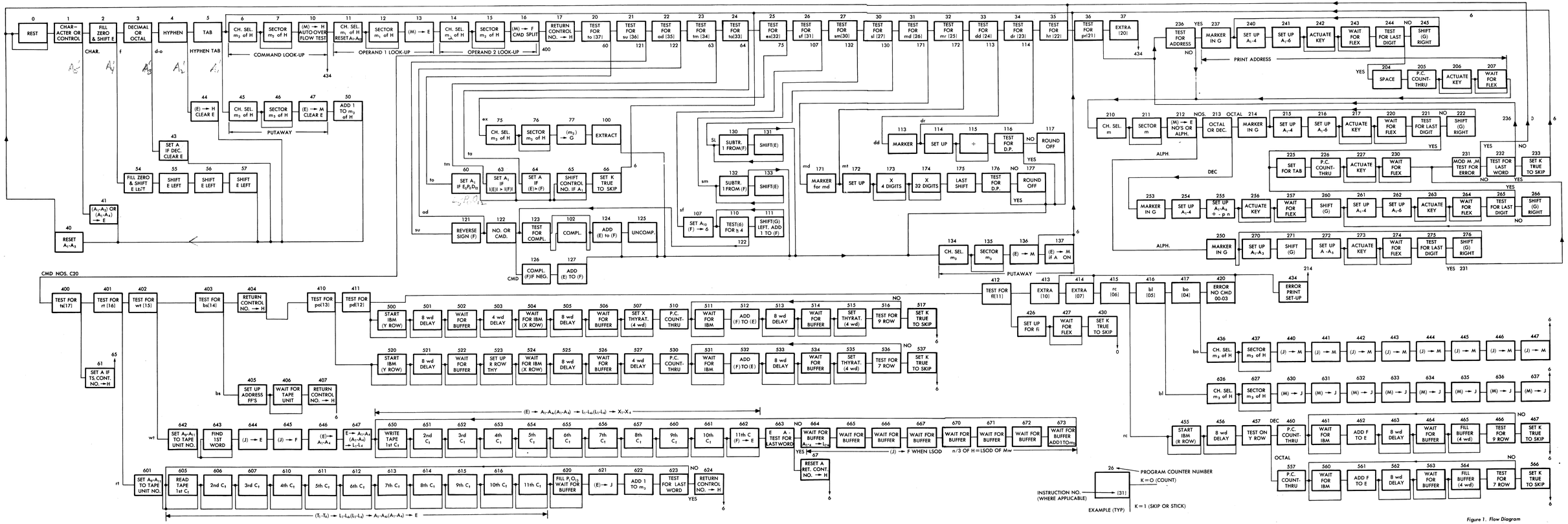


Figure 1. Flow Diagram

FLOW

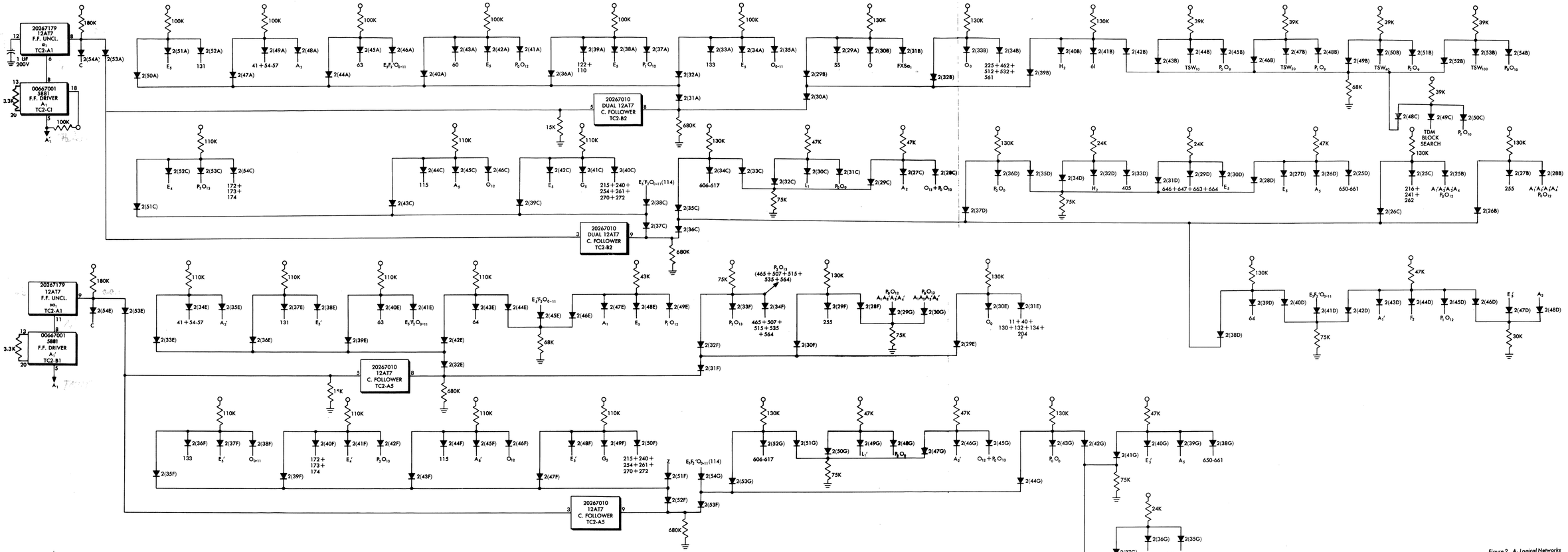


Figure 2. A_1 Logical Networks

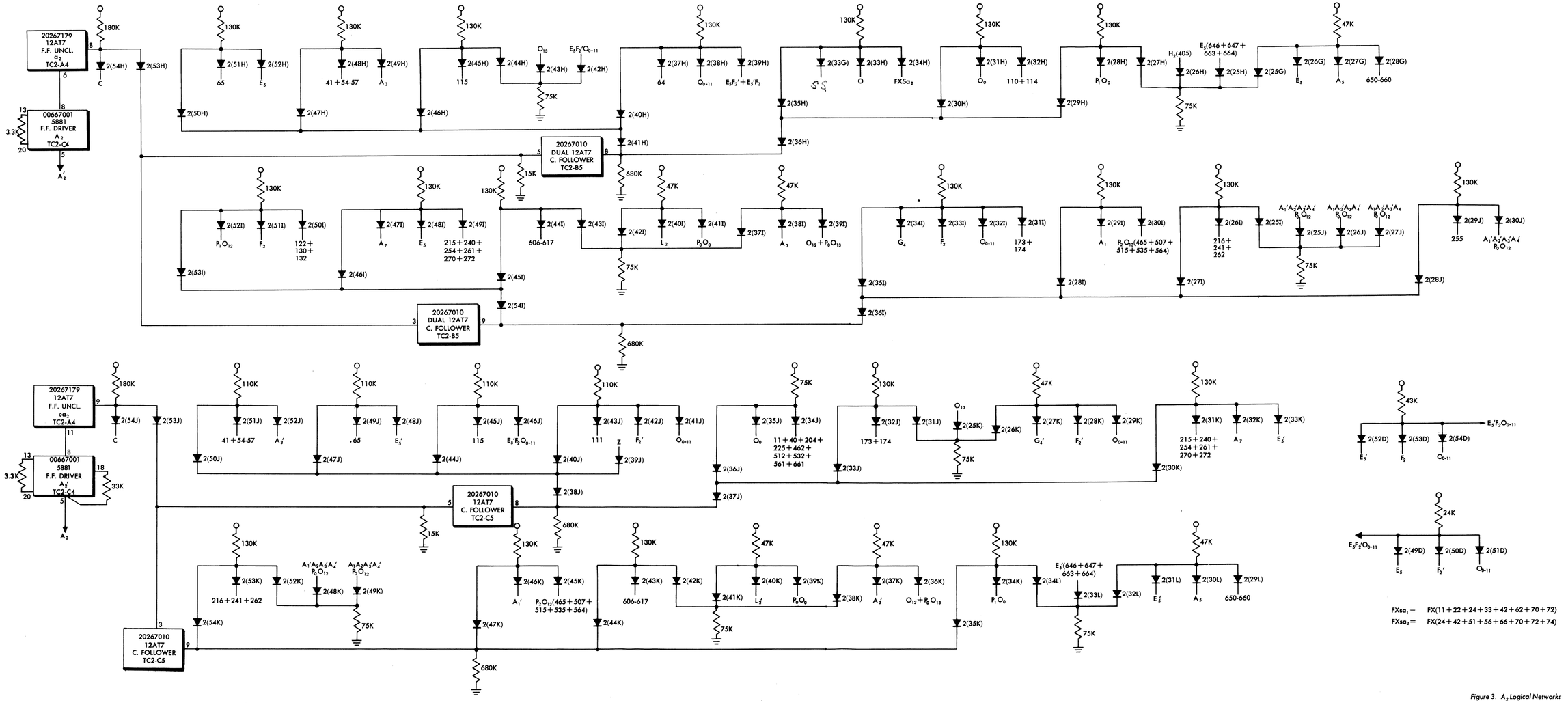


Figure 3. A_2 Logical Networks

A2

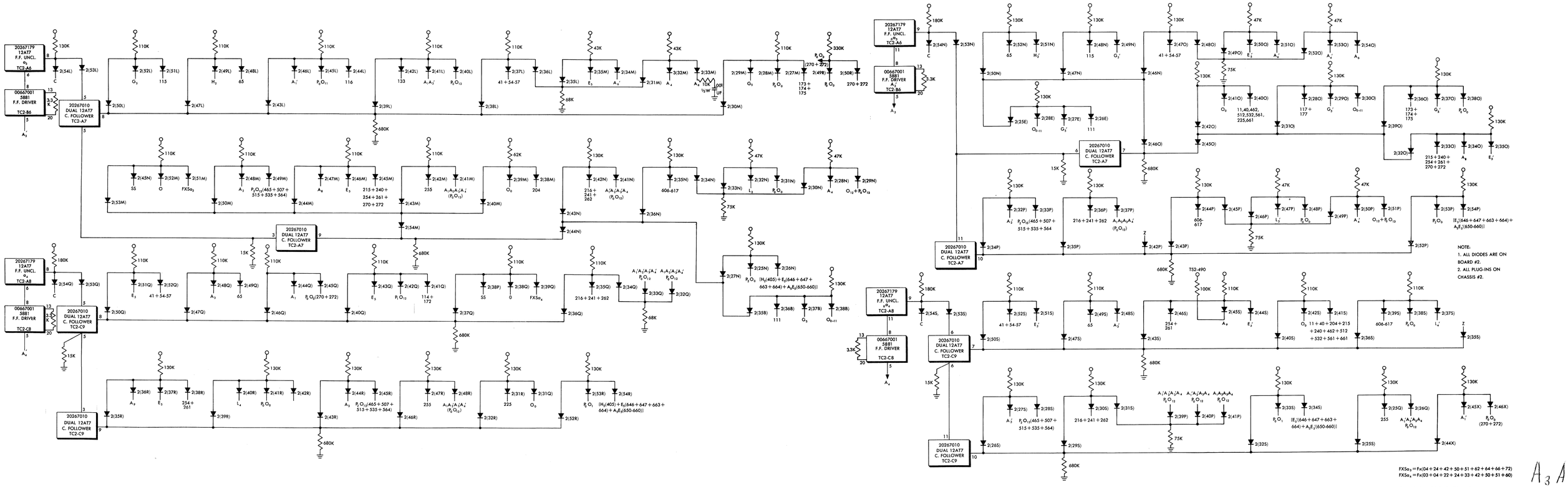

20267179 12A7 F.F. UNCL. a_3 TC2-A6
00667001 5881 F.F. DRIVER TC2-B6
20267010 DUAL 12A7 C. FOLLOWER TC2-A7
20267179 12A7 F.F. UNCL. a_4 TC2-A8
00667001 5881 F.F. DRIVER TC2-C8
20267010 DUAL 12A7 C. FOLLOWER TC2-C9
NOTE:
1. ALL DIODES ARE ON BOARD #2.
2. ALL PLUG-INS ON CHASSIS #2.
FXSa3 = Fx(04 + 24 + 42 + 50 + 51 + 62 + 64 + 66 + 72)
FXSa4 = Fx(03 + 04 + 22 + 24 + 33 + 42 + 50 + 51 + 60)
A3 A4

Figure 4. A_3 and A_4 Logical Networks

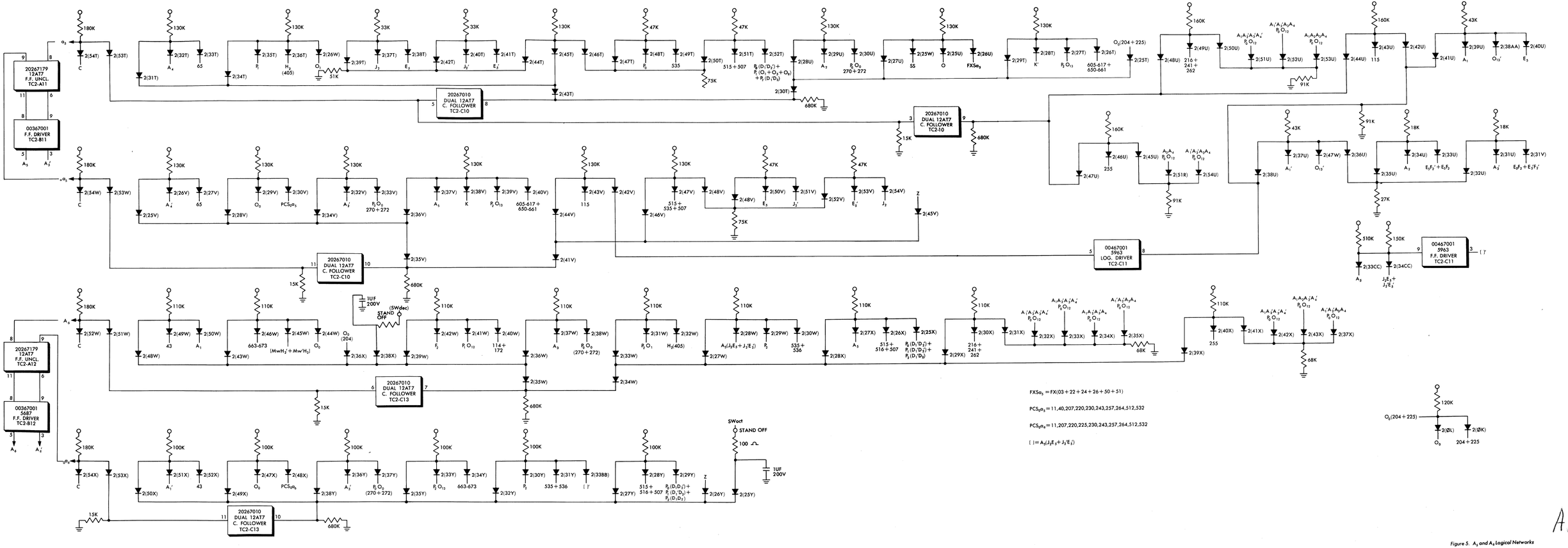


Figure 5. A_5 and A_6 Logical Networks

A5 A6

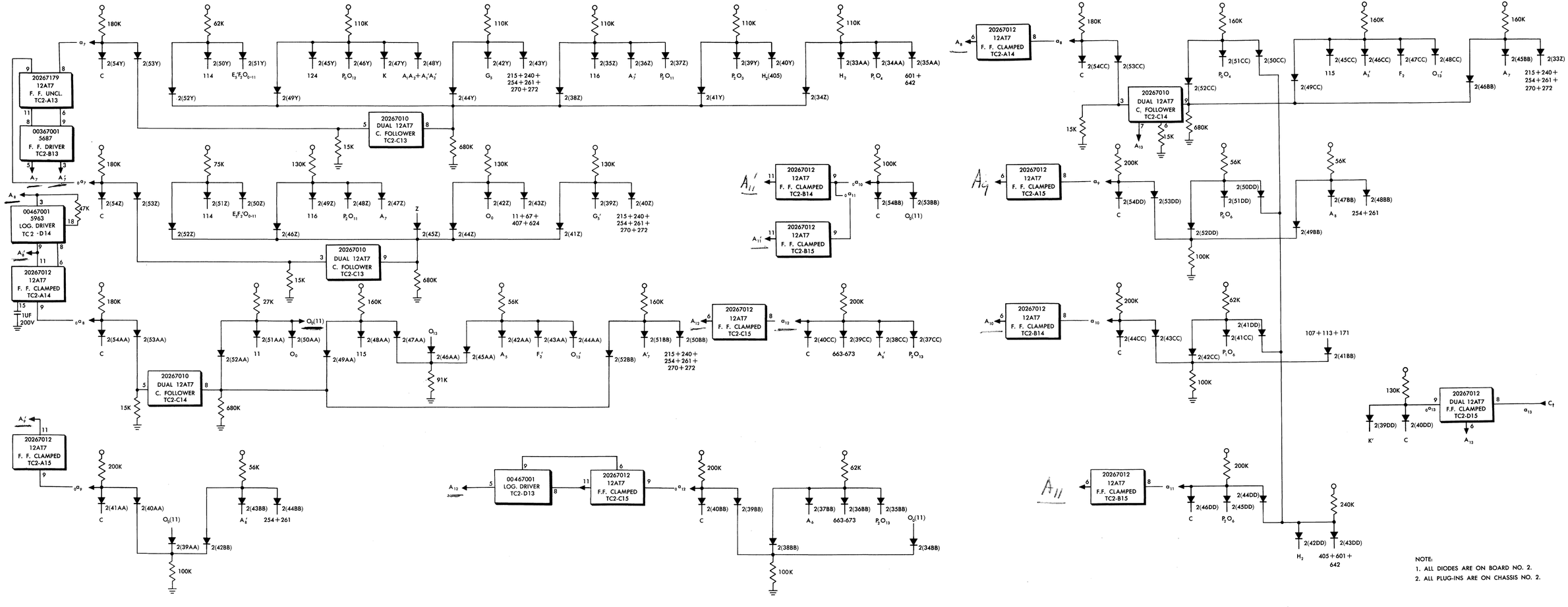


Figure 6. A_7 Through A_{12} Logical Networks

$A_7 \Leftrightarrow A_{12}$

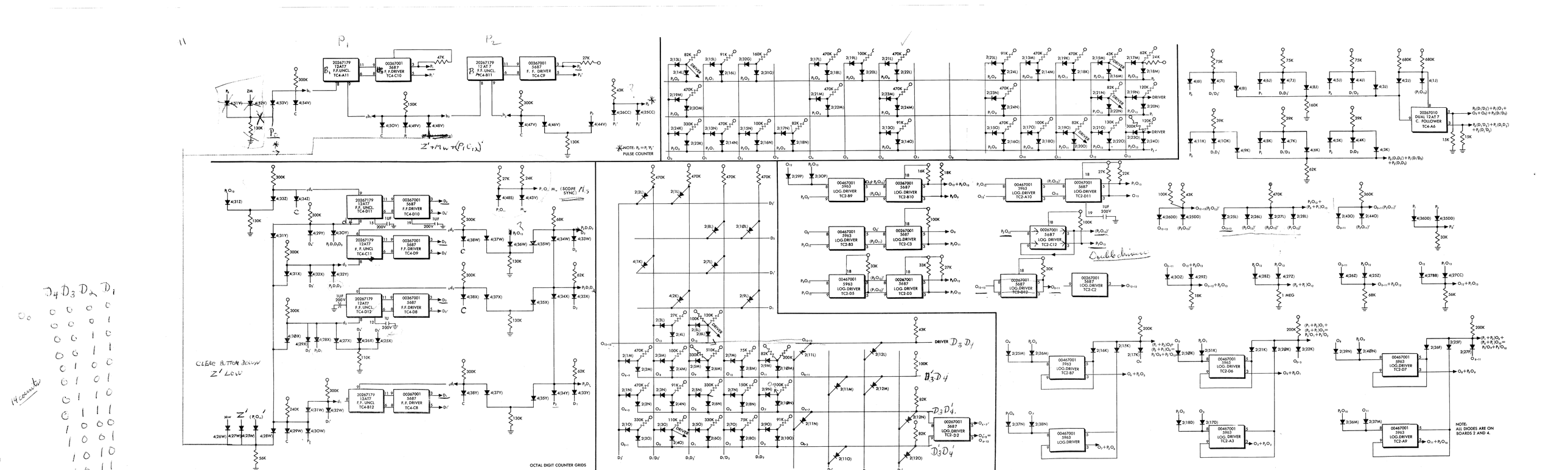


Figure 7. Pulse and Digit Counter Logical Networks

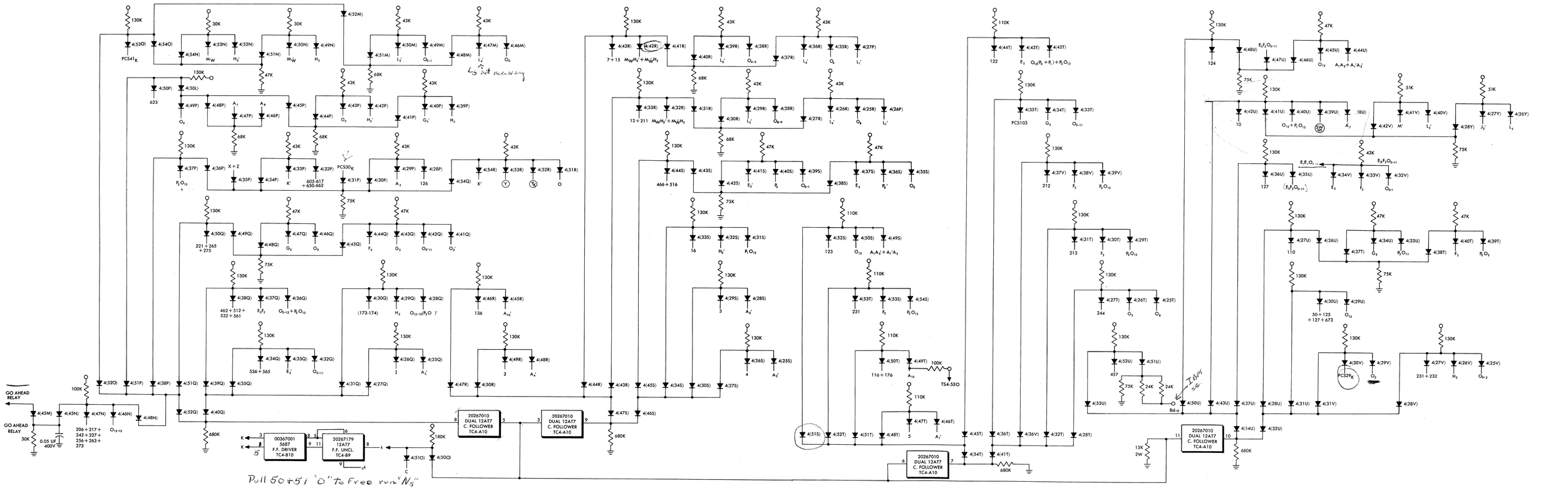
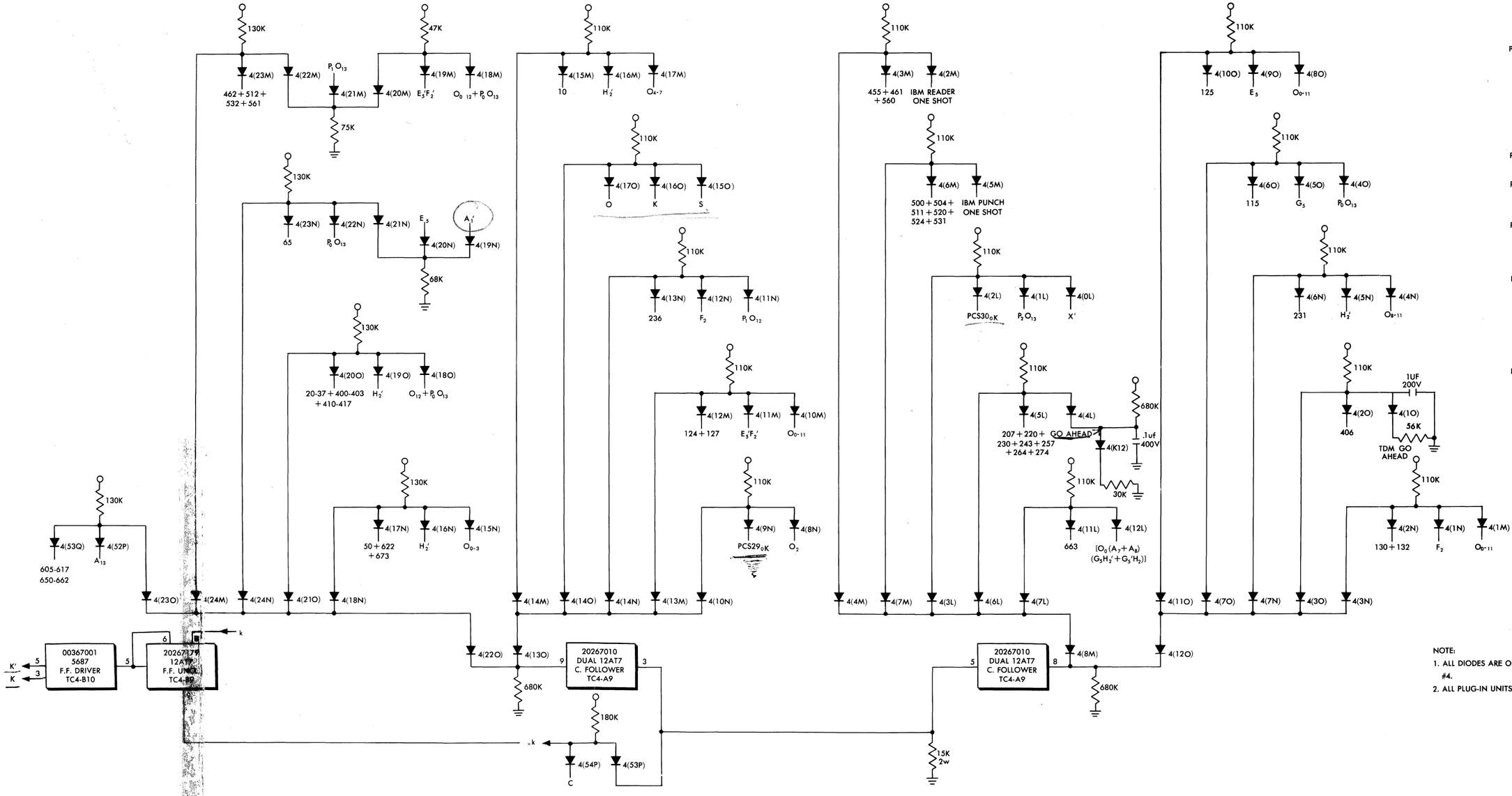


Figure 8. K Logical Networks



$PCS29_k$ = 10,20-37,57,66,100, 111,115,117,131,133,137, 177,207,220,222,230, 234,243,245,257,264,266, 274,276,400-404,407, 410-417,420,430,447,467, 504,517,524,537,566, 624,637.

$PCS30_k$ = 47,124,405,460, 510,530,557,621,647,672.

PCS41 = 46,76,135,437, 464,502,506,514,522,526, 534,563,620,627,643.

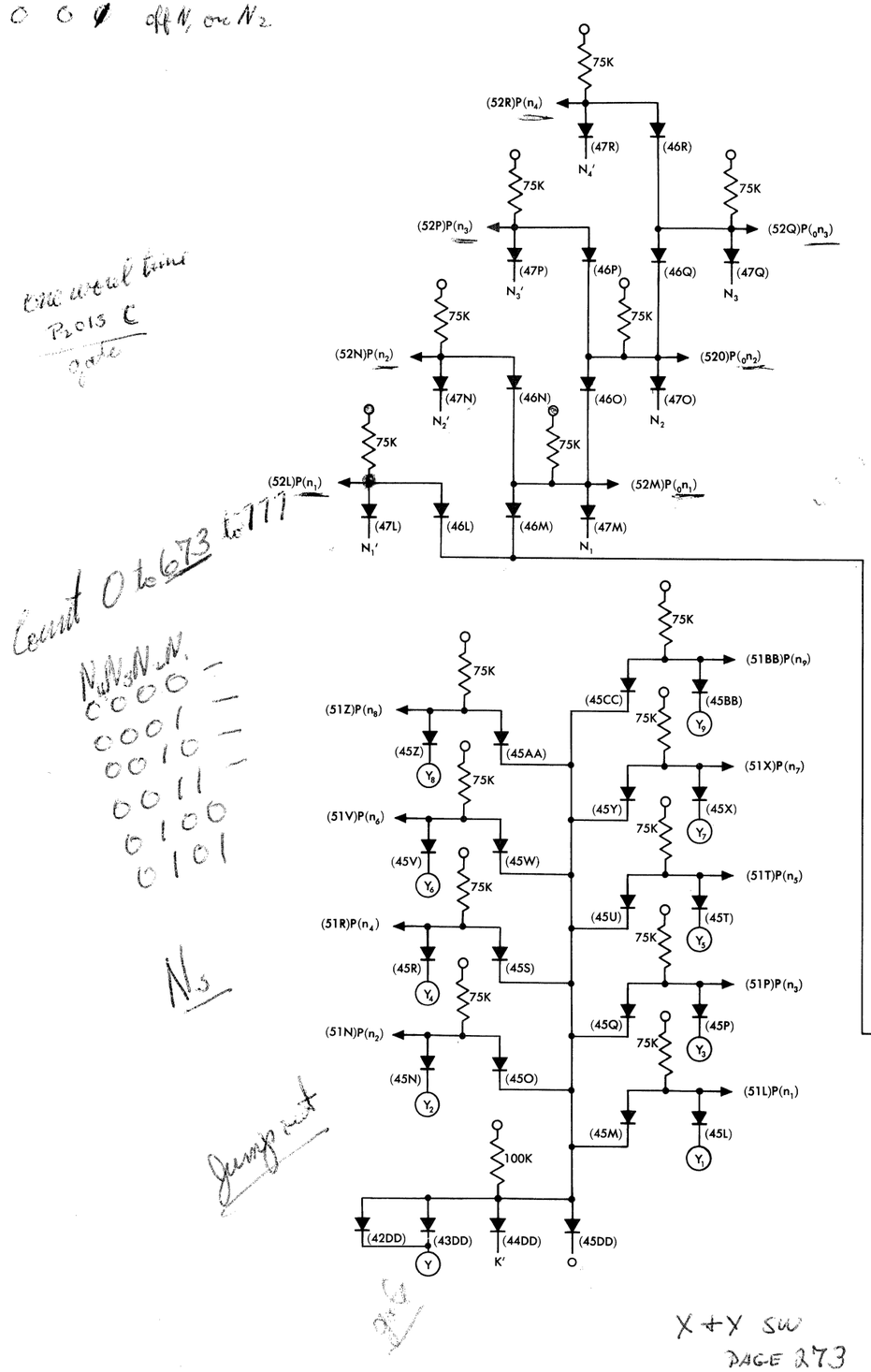
PCS103 = 427,456,463, 465,501,503,505,507, 513,515,521,523,525,527, 533,535,562,564.

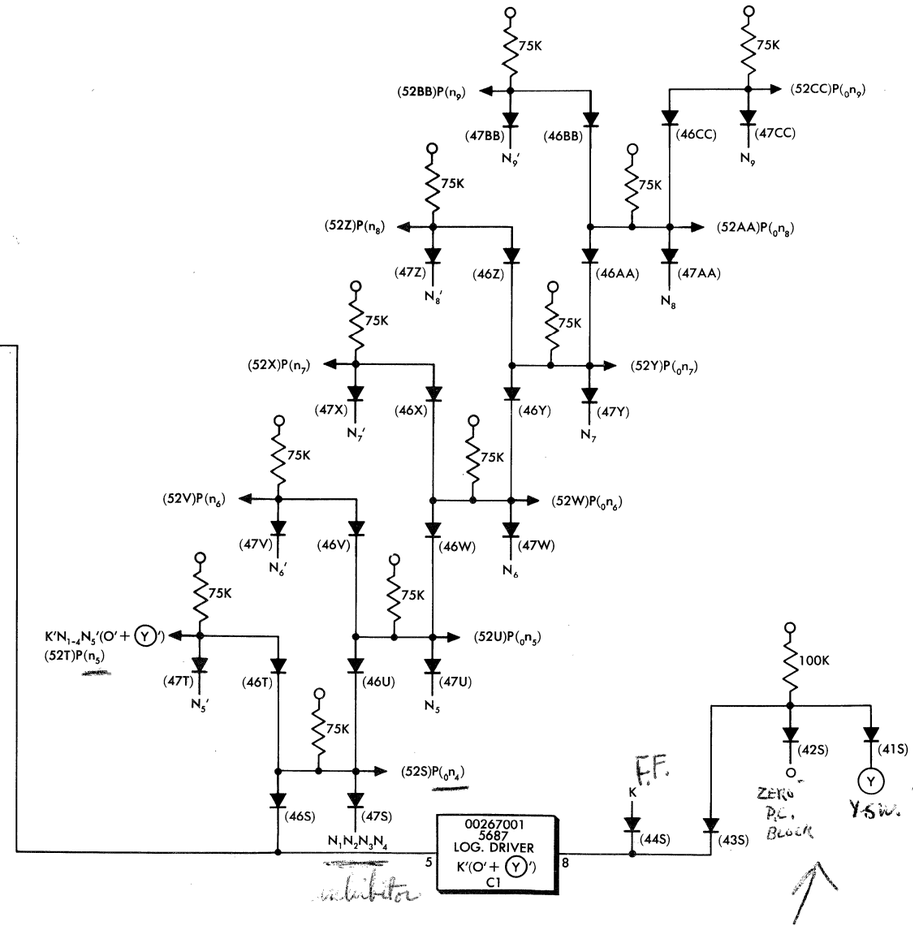
$PCS29_0k$ = 6,45,54,75,107, 113,114,121,122, 126,134,171,172,204, 206,210,215,217,225, 227,240,242, 253,256, 261,263,270,273,405, 426,436,557,626.

$PCS30_{0k}$ = 7,11,12,15,16, 46,76,111,135,173,174, 211,427,437,456,463,464, 465,501,502,503,505,506, 507,513,514,515,521,522,523, 525,526,527,533,534,535, 562,563,564,620,627,643.

NOTE:
1. ALL DIODES ARE ON BOARD #4.
2. ALL PLUG-IN UNITS ON TC4.

Figure 9. ok Logical Networks





$n_1 = [K'N_1'(O' + (Y)') + (Y)(Y_1)K'O + K(PCSn_1)]X'P_2O_{13}C$

${}_0n_1 = [K'N_1(O' + (Y)') + X + Z + K(PCS_0n_1)]P_2O_{13}C$

$n_2 = [K'N_1N_2'(O' + (Y)') + (Y)(Y_2)K'O + K(PCSn_2)]X'P_2O_{13}C$

${}_0n_2 = [K'N_1N_2(O' + (Y)') + X + Z + K(PCS_0n_2)]P_2O_{13}C$

$n_3 = [K'N_1N_2N_3'(O' + (Y)') + (Y)(Y_3)K'O + K(PCSn_3)]X'P_2O_{13}C$

${}_0n_3 = [K'N_1N_2N_3(O' + (Y)') + X + Z + K(PCS_0n_3)]P_2O_{13}C$

$n_4 = [K'N_1N_2N_3N_4'(O' + (Y)') + (Y)(Y_4)K'O + K(PCSn_4)]X'P_2O_{13}C$

${}_0n_4 = [K'N_1N_2N_3N_4(O' + (Y)') + X + Z + K(PCS_0n_4)]P_2O_{13}C$

$n_5 = [K'N_1N_2N_3N_4N_5'(O' + (Y)') + (Y)(Y_5)K'O + K(PCSn_5)]X'P_2O_{13}C$

${}_0n_5 = [K'N_1N_2N_3N_4N_5(O' + (Y)') + X + Z + K(PCS_0n_5)]P_2O_{13}C$

$n_6 = [K'N_1N_2N_3N_4N_5N_6'(O' + (Y)') + (Y)(Y_6)K'O + K(PCSn_6)]X'P_2O_{13}C$

${}_0n_6 = [K'N_1N_2N_3N_4N_5N_6(O' + (Y)') + X + Z + K(PCS_0n_6)]P_2O_{13}C$

$n_7 = [K'N_1N_2N_3N_4N_5N_6N_7'(O' + (Y)') + (Y)(Y_7)K'O + K(PCSn_7)]X'P_2O_{13}C$

${}_0n_7 = [K'N_1N_2N_3N_4N_5N_6N_7(O' + (Y)') + X + Z + K(PCS_0n_7)]P_2O_{13}C$

$n_8 = [K'N_1N_2N_3N_4N_5N_6N_7N_8'(O' + (Y)') + (Y)(Y_8)K'O + K(PCSn_8)]X'P_2O_{13}C$

${}_0n_8 = [K'N_1N_2N_3N_4N_5N_6N_7N_8(O' + (Y)') + X + Z + K(PCS_0n_8)]P_2O_{13}C$

$n_9 = [K'N_1N_2N_3N_4N_5N_6N_7N_8N_9'(O' + (Y)') + (Y)(Y_9)K'O + K(PCSn_9)]X'P_2O_{13}C$

${}_0n_9 = [K'N_1N_2N_3N_4N_5N_6N_7N_8N_9(O' + (Y)') + X + Z + K(PCS_0n_9)]P_2O_{13}C$

$PCSn_1 = 26,60,64,110,222,266,400,$
$466,516,536,642$

$PCS_0n_1 = 27,35,37,41,43,57,67,$
$111,117,123,125,127,131,133,$
$137,177,231,233,245,407,411,$
$413,417,447,467,517,537,565,$
$637,673$

$PCSn_2 = 110,624$

$PCS_0n_2 = 2,16,37,43,57,63,102,$
$116,117,127,132,176,177,212,$
$222,236,266,276,403,413,466,$
$516,536,623,673$

$PCSn_3 = 2,10,60,61,63,100,102,$
$122,130,132,221,222,232,233,$
$403,412,413,420,601,623,663$

$PCS_0n_3 = 16,27,35,44,57,236,245,$
$266,275,276,404,466,516,536,565$

$PCSn_4 = 2,25,27,100,124,125,127,$
$222,404,420$

$PCS_0n_4 = 16,35,50,57,110,136,137,$
$231,233,410,411,412,416,430,$
$517,537,637$

$PCSn_5 = 10,100,102,110,116,117,250,$
$400,410,412,413,414,416,417$

$PCS_0n_5 = 26,33,34,35,66,67,123,$
$136,137,222,231,233,236,430,$
$434,467,537,566,623,624$
$637,673$

$PCSn_6 = 1,2,3,4,5,20,23,$
$24,25,31,32,212,213,400,$
$402,415$

$PCS_0n_6 = 40,66,67,176,177,244,265,$
$275,447,467,566$

$PCSn_7 = 21,22,26,27,30,31,32,$
$33,34,410,411,457$

$PCS_0n_7 = 136,137,517,537,566$

$PCSn_8 = 36,401,402,416,434$

$PCS_0n_8 = 231,233,624,637,663$

$PCSn_9 = 10,16,37$

$PCS_0n_9 = 400,407,430,434,447,467,517,$
$537,566,624,637,663$

NOTES:
1. ALL DIODES ARE ON BOARD #3
2. ALL PLUG-INS ARE ON CHASSIS #3

Figure 10 (1 of 2). Program Counter Grid Networks

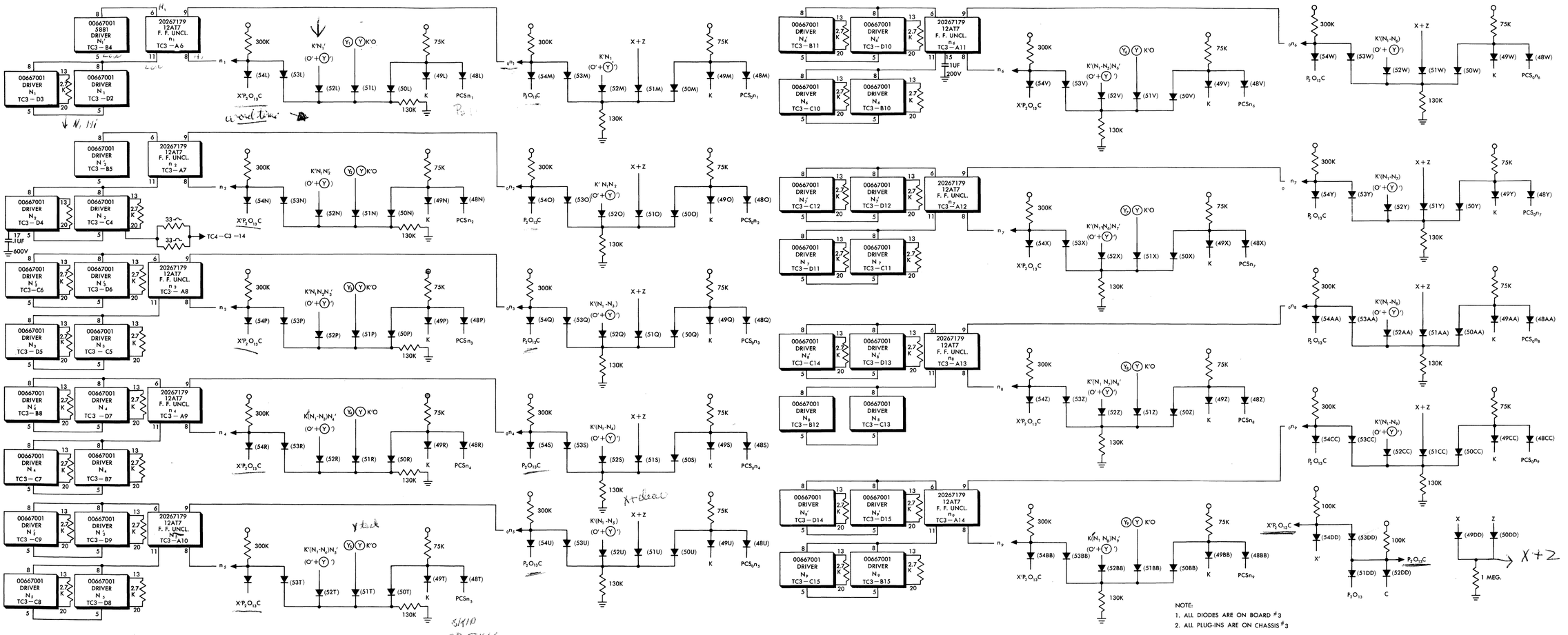


Figure 10(2 of 2). Program Counter Grid Networks

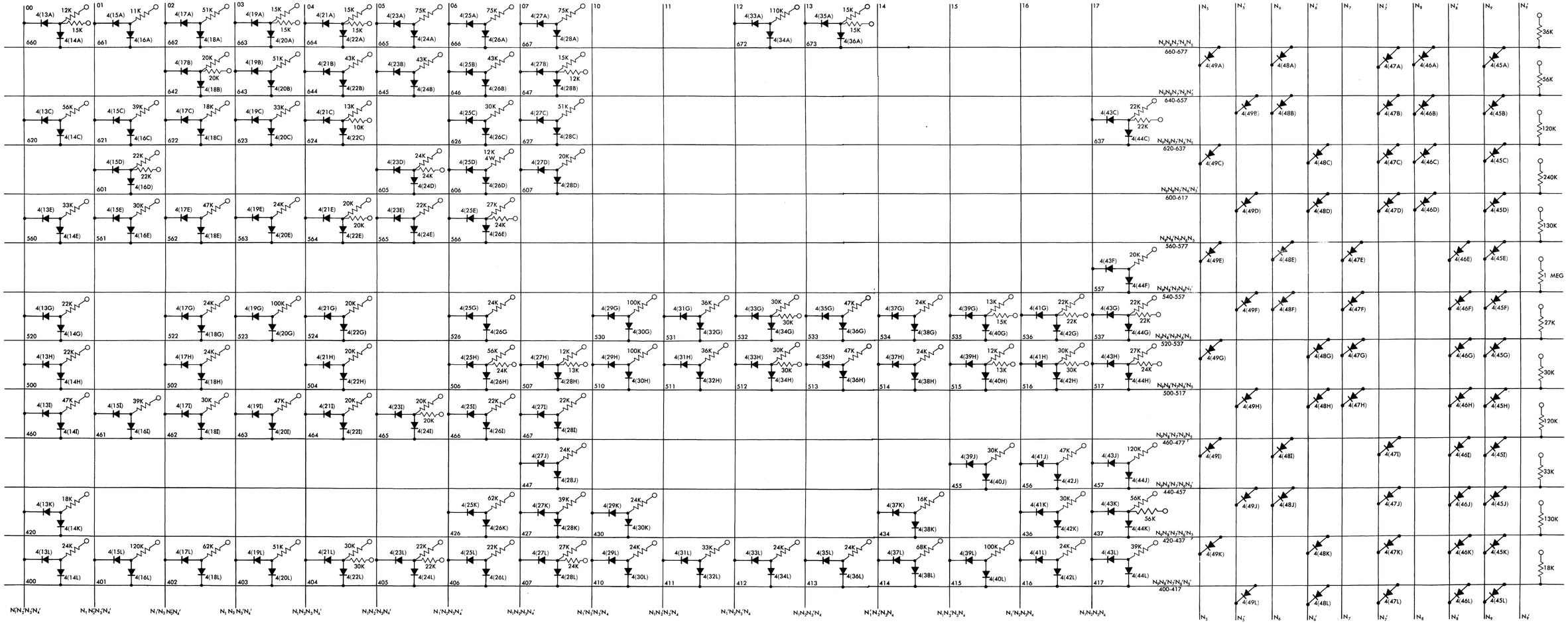


Figure 11 (1 of 3). Program Counter Matrix Networks

P.C. BLOCK MATRIX

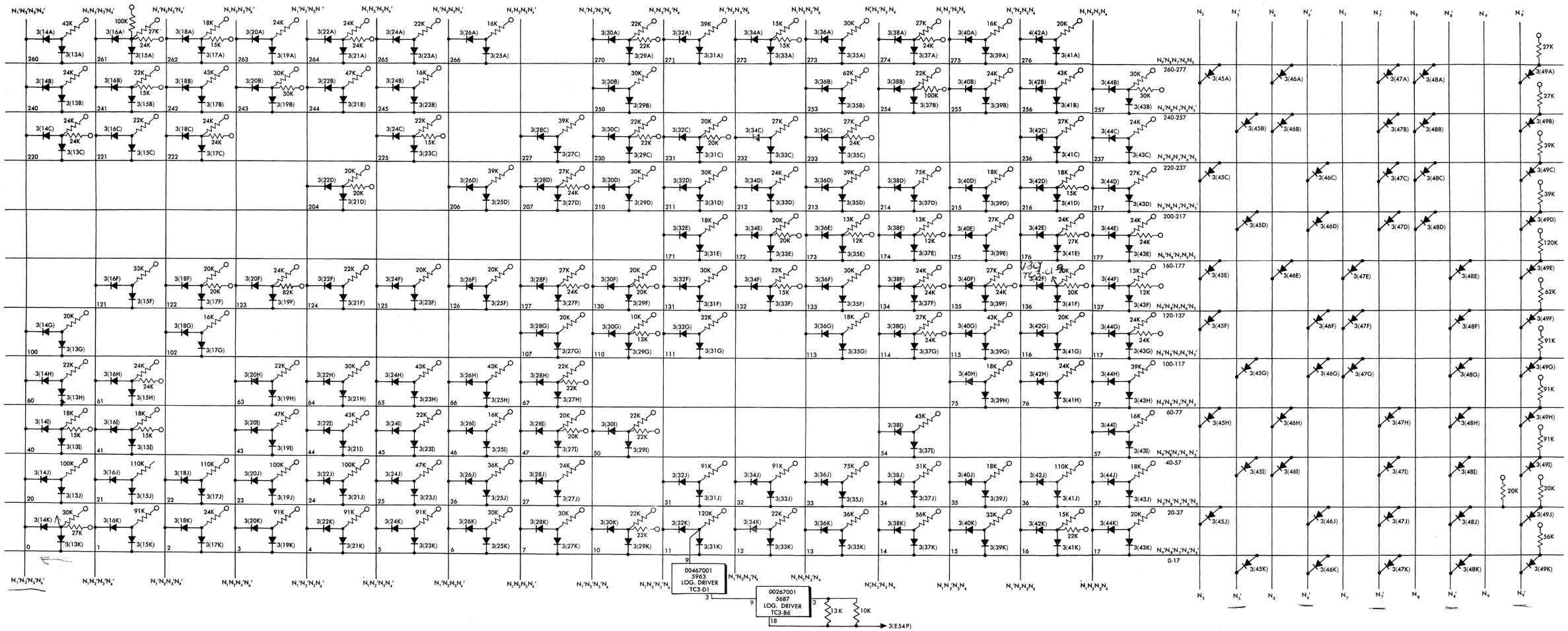
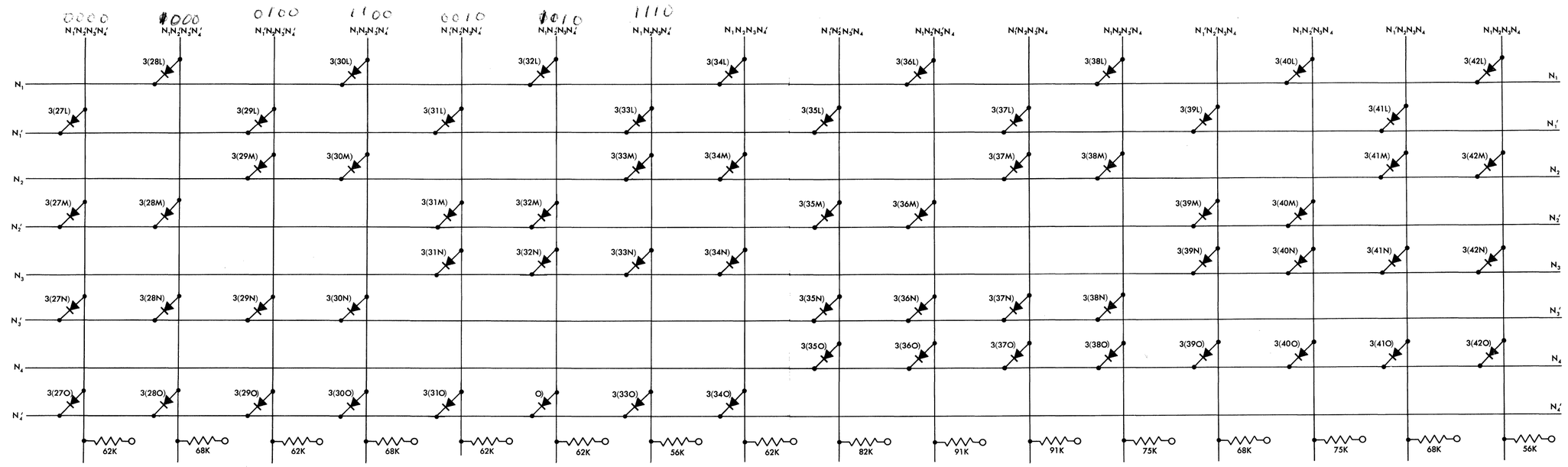


Figure 11 (2 of 3). Program Counter Matrix Networks

PC
BLOCK
235 MATRIX



$N_9N_8N_7N_6N_5 = 20\text{-}37$
(USED IN PCS29$_K$ PCS6)
$N_9N_8N_7N_6N_5' = 200\text{-}217$
(USED IN PCS2)
$N_9N_8N_7N_6'N_5 = 220\text{-}237$
(USED IN PCS2)
$N_9N_8N_7N_6'N_5' = 240\text{-}257$
(USED IN PCS6)
$N_9N_8N_7N_6N_5 = 260\text{-}277$
(USED IN PCS6)
$N_9N_8N_7'N_6N_5' = 400\text{-}417$
(USED IN PCS3 & PCS6)
$N_9N_8'N_7N_6N_5' = 500\text{-}517$
(USED IN PCS6)
$N_9N_8'N_7'N_6N_5' = 440\text{-}457$
(USED IN PCS6)
$N_9N_8'N_7N_6'N_5 = 520\text{-}537$
(USED IN PCS6)
$N_9N_8'N_7N_6N_5 = 660\text{-}673$
(USED IN PCS3)

NOTE:
ALL DIODES ON BOARDS 3 & 4.

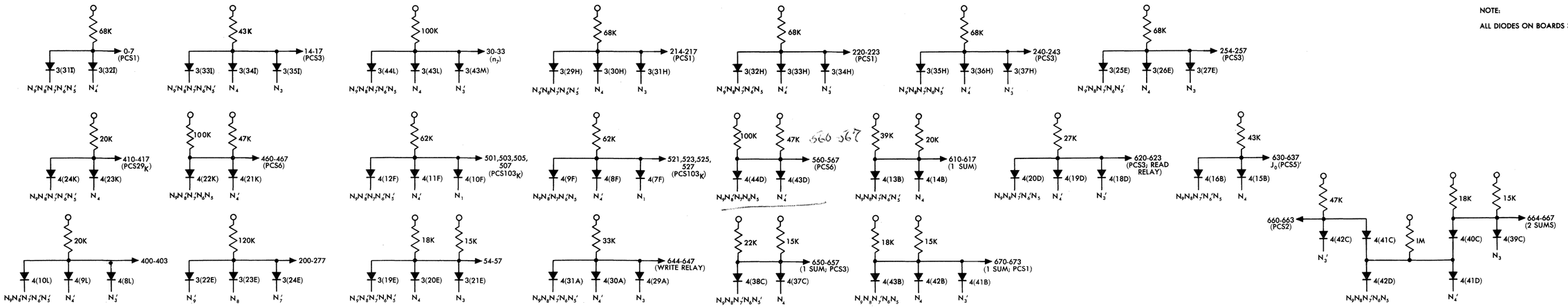


Figure 11 (3 of 3). Program Counter Matrix Networks

PC SUMS

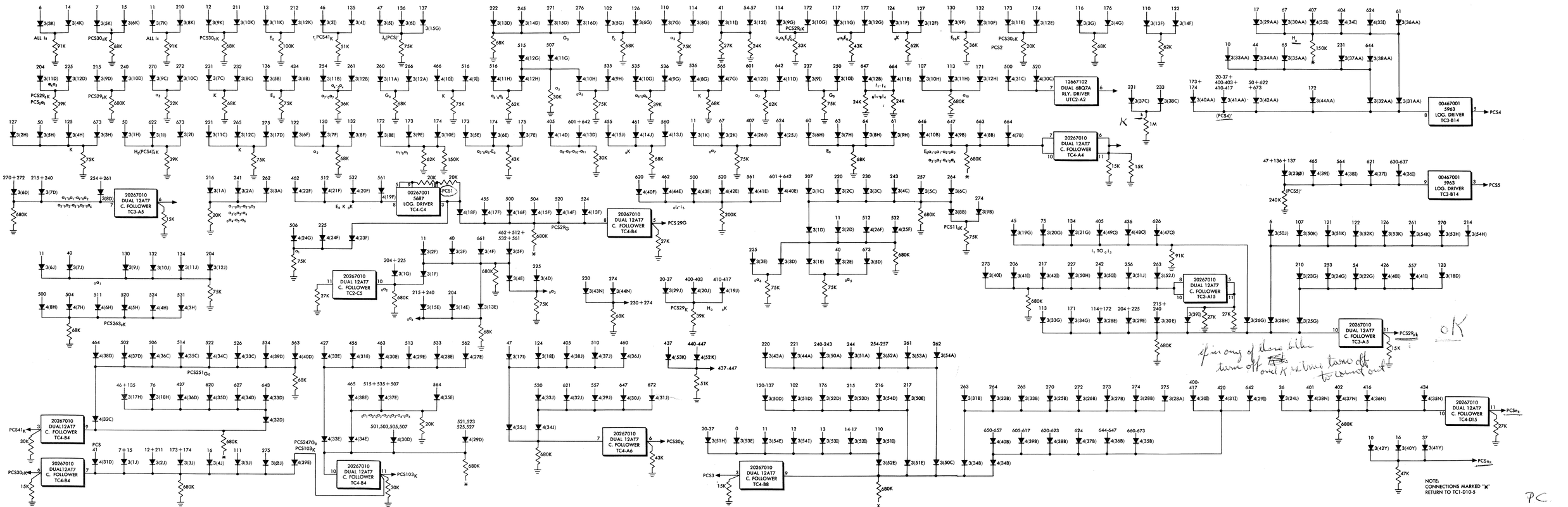

20267010 DUAL 12A77 C. FOLLOWER TC3-A5
20267010 DUAL 12A77 C. FOLLOWER TC2-C5
00267001 5687 LOG. DRIVER TC4-C4
20267010 DUAL 12A77 C. FOLLOWER TC4-B4
12667102 DUAL 68074 RLY. DRIVER UTC2-A2
00467001 5963 LOG. DRIVER TC3-B14
20267010 DUAL 12A77 C. FOLLOWER TC4-A4
20267010 DUAL 12A77 C. FOLLOWER TC4-A6
20267010 DUAL 12A77 C. FOLLOWER TC4-B8
20267010 DUAL 12A77 C. FOLLOWER TC4-D15
NOTE: CONNECTIONS MARKED "*" RETURN TO TC1-D10-5

Figure 12(1 of 3). Program Counter Sum Networks

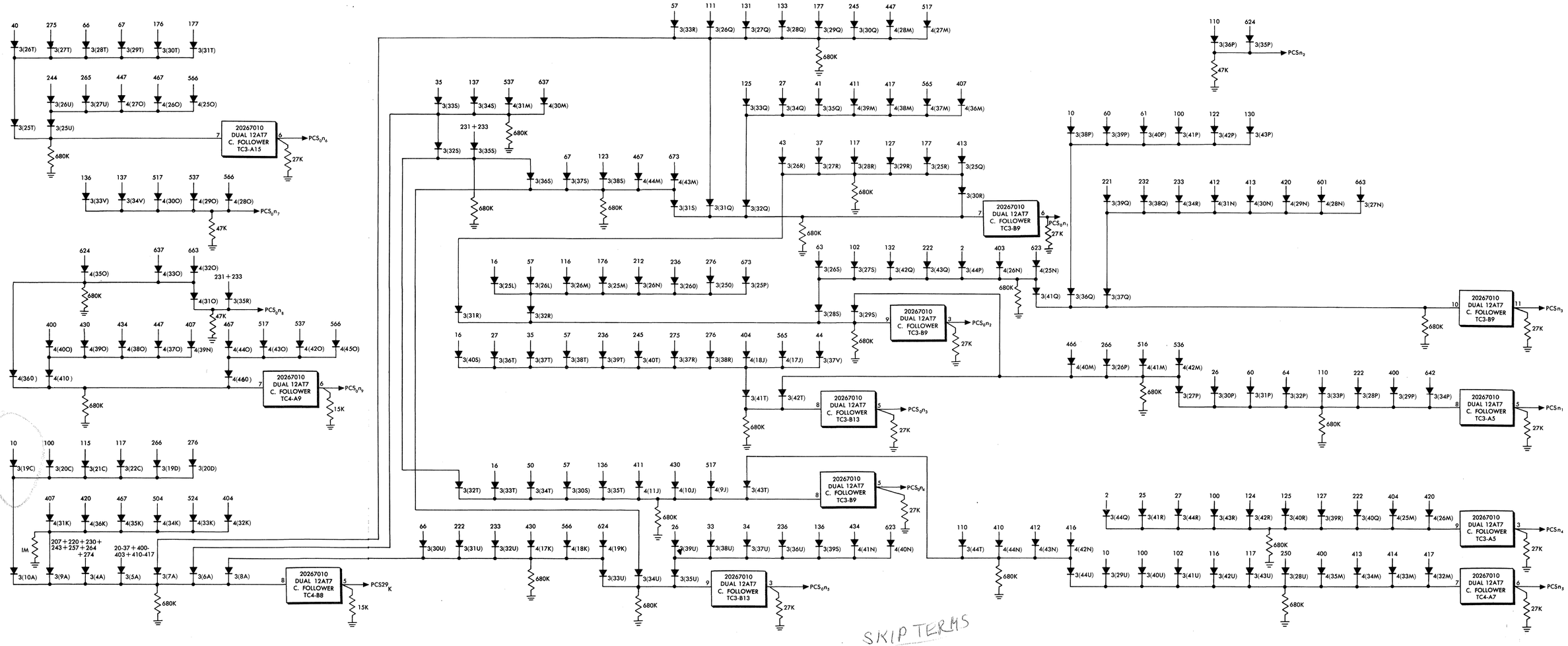
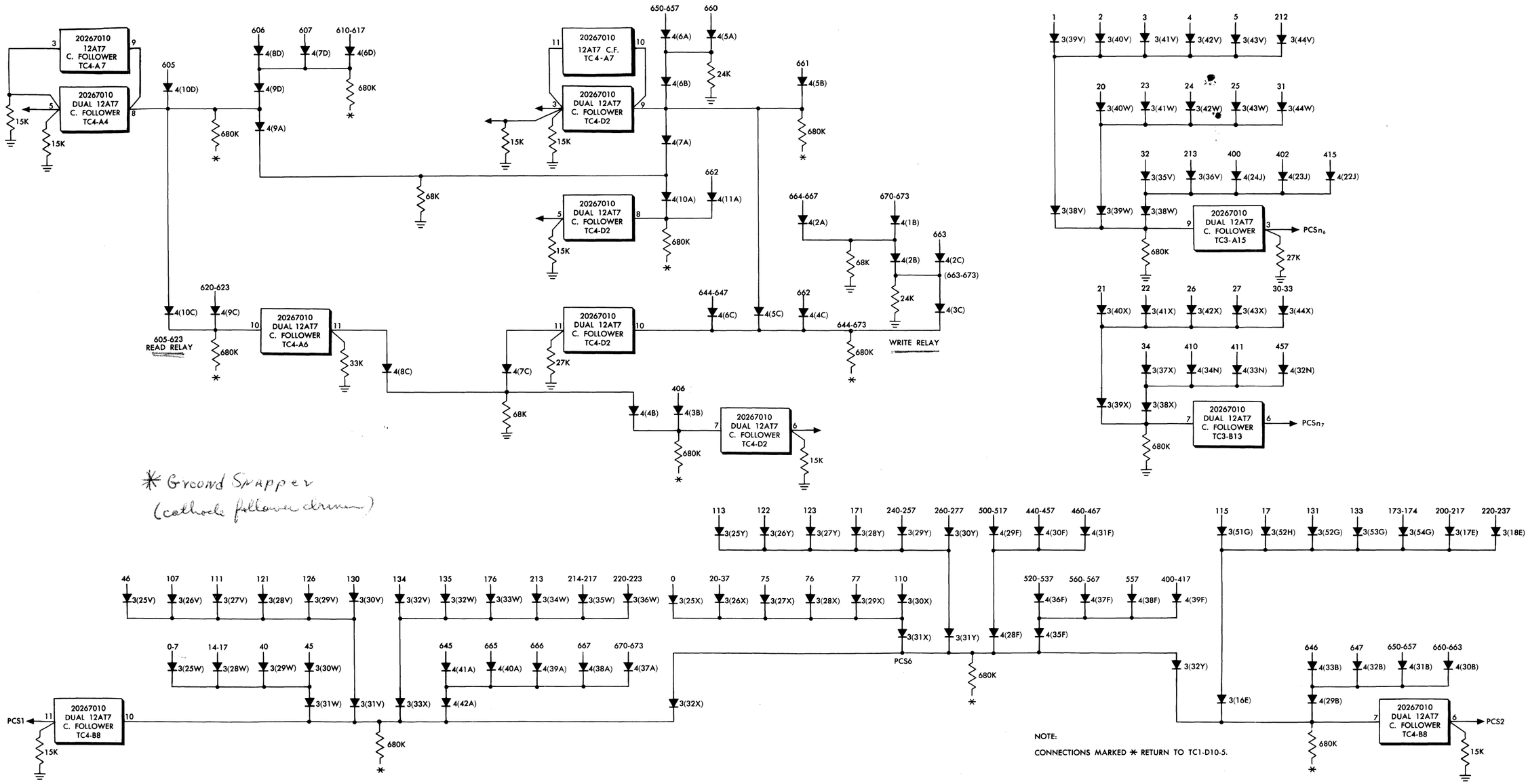


Figure 12 (2 of 3). Program Counter Sum Networks



* Ground Strapper (cathode follower drum)

Figure 12 (3 of 3). Program Counter Sum Networks

PC SUMS

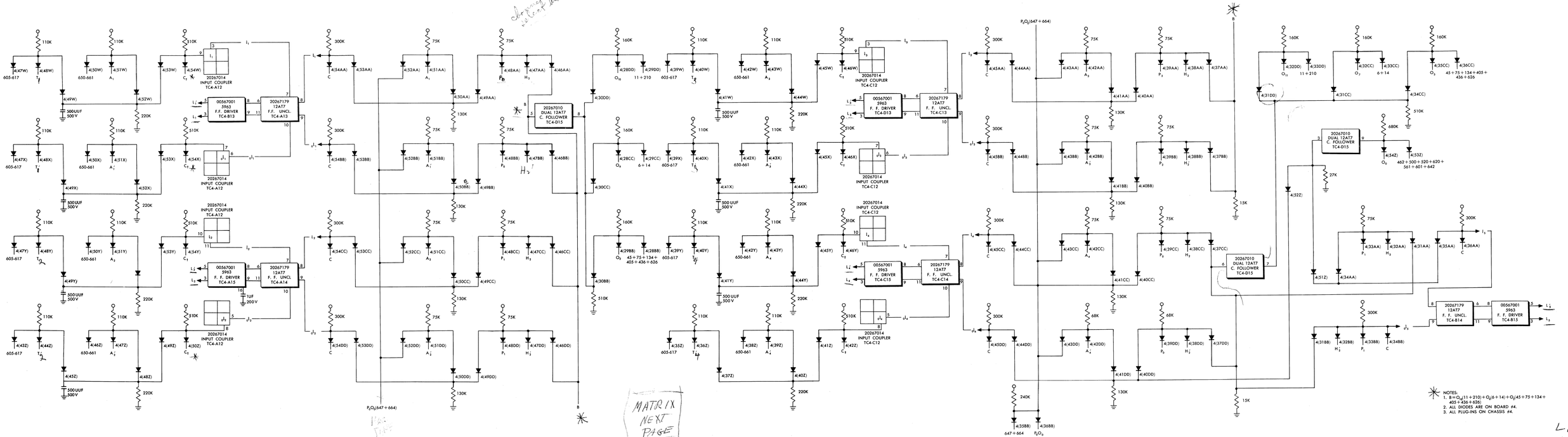
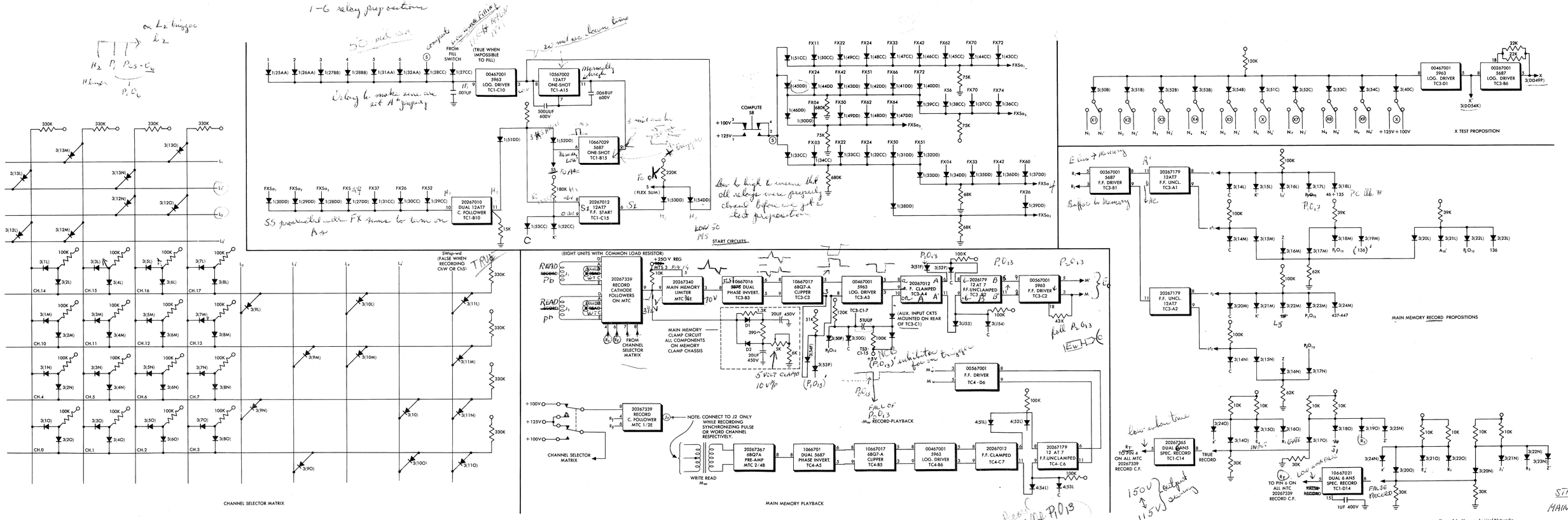

20267014 INPUT COUPLER TC4-A12
00567001 5963 F.F. DRIVER TC4-B13
20267179 12AT7 F.F. UNCL. TC4-A13
20267010 DUAL 12AT7 C. FOLLOWER TC4-D15
00567001 5963 F. F. DRIVER TC4-A15
20267179 12AT7 F. F. UNCL. TC4-A14
20267179 12AT7 F. F. UNCL. TC4-C13
00567001 5963 F. F. DRIVER TC4-C15
20267179 12AT7 F. F. UNCL. TC4-C14
00567001 5963 F. F. DRIVER TC4-B14
00567001 5963 F. F. DRIVER TC4-B15
MATRIX NEXT PAGE
NOTES:
1. B = O10(11 + 210) + O9(6 + 14) + O8(45 + 75 + 134 + 405 + 436 + 626)
2. ALL DIODES ARE ON BOARD #4.
3. ALL PLUG-INS ON CHASSIS #4.

Figure 13. L_1 Through L_4 Channel Selector Logical Networks


CHANNEL SELECTOR MATRIX
MAIN MEMORY PLAYBACK
MAIN MEMORY RECORD PROPOSITIONS
X TEST PROPOSITION
START CIRCUITS
20267339 RECORD CATHODE FOLLOWERS ON MTC
20267340 MAIN MEMORY LIMITER MTC 36E
20267339 RECORD C. FOLLOWER MTC 1/2E
20267365 DUAL 6AN5 SPEC. RECORD TC1-C14
10667021 DUAL 6 AN5 SPEC. RECORD TC1-D14
20267010 DUAL 12AT7 C. FOLLOWER TC1-B10
20267012 12AT7 F.F. START TC1-C15
20267179 12 AT 7 F.F.UNCLAMPED TC4- C6
00467001 5963 LOG. DRIVER TC3-D1
00367001 5687 F.F. DRIVER TC3-B1
00567001 5963 F.F. DRIVER TC3-C2
00567001 F.F. DRIVER TC4 - D6
10667016 DUAL PHASE INVERT. TC3-B3
10667017 6BQ7-A CLIPPER TC3-C3
00467001 5963 LOG. DRIVER TC3-A3
20267012 F.F. CLAMPED TC3-A4
10667017 6BQ7-A CLIPPER TC4-B5
00467001 5963 LOG. DRIVER TC4-B6
20267012 F.F. CLAMPED TC4-C7
1066701 DUAL 5687 PHASE INVERT TC4-A5
20267367 6BQ7A PRE-AMP MTC 2/4B
10567002 12AT7 ONE-SHOT TC1-A15
10667029 5687 ONE-SHOT TC1-B13
20267179 12AT7 F.F. UNCL. TC3-A1
20267179 12AT7 F.F. UNCL. TC3-A2
(EIGHT UNITS WITH COMMON LOAD RESISTOR)
+250 V REG
MAIN MEMORY CLAMP CIRCUIT ALL COMPONENTS ON MEMORY CLAMP CHASSIS
5 VOLT CLAMP 10 V/P
FROM CHANNEL SELECTOR MATRIX
CHANNEL SELECTOR MATRIX
WRITE READ
NOTE: CONNECT TO J2 ONLY WHILE RECORDING SYNCHRONIZING PULSE OR WORD CHANNEL RESPECTIVELY.
FAILS OF -Mw RECORD-PLAYBACK
AUX. INPUT CKTS MOUNTED ON REAR OF TC3-C1)
TRUE RECORD
FALSE RECORD
TO PIN 4 ON ALL MTC 20267339 RECORD C.F.
TO PIN 6 ON ALL MTC 20267339 RECORD C.F.
+100V
+125V
+1125V+100V
COMPUTE S8
FROM FILL SWITCH
(TRUE WHEN IMPOSSIBLE TO FILL)
FLEX SUM
1UF 400V
Figure 14. Memory Logical Networks

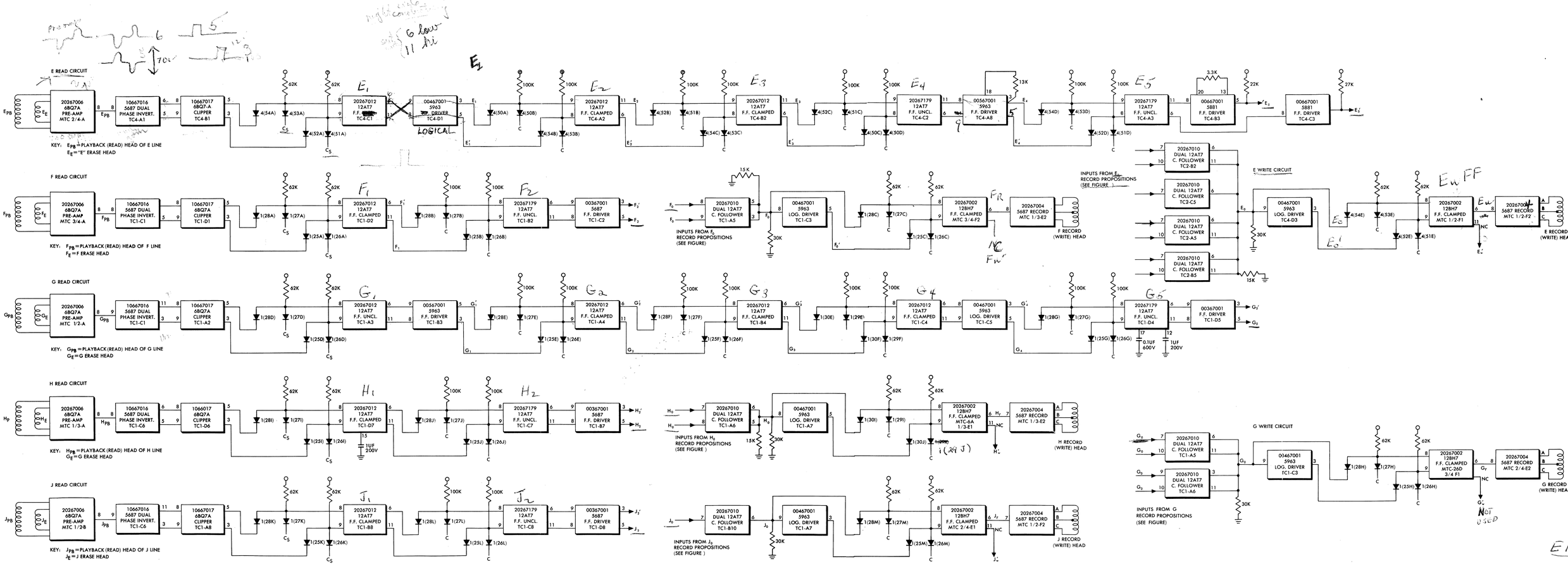

E READ CIRCUIT
KEY: E_{PB} = PLAYBACK (READ) HEAD OF E LINE
E_E = "E" ERASE HEAD
F READ CIRCUIT
KEY: F_{PB} = PLAYBACK (READ) HEAD OF F LINE
F_E = F ERASE HEAD
G READ CIRCUIT
KEY: G_{PB} = PLAYBACK (READ) HEAD OF G LINE
G_E = G ERASE HEAD
H READ CIRCUIT
KEY: H_{PB} = PLAYBACK (READ) HEAD OF H LINE
G_E = G ERASE HEAD
J READ CIRCUIT
KEY: J_{PB} = PLAYBACK (READ) HEAD OF J LINE
J_E = J ERASE HEAD
20267006 68Q7A PRE-AMP MTC 2/4-A
10667016 5687 DUAL PHASE INVERT. TC4-A1
10667017 68Q7-A CLIPPER TC4-B1
20267012 12AT7 F.F. CLAMPED TC4-C1
00467001 5963 DRIVER TC4-D1
20267179 12AT7 F.F. UNCL. TC4-A2
00567001 5963 F.F. DRIVER TC4-B2
20267179 12AT7 F.F. UNCL. TC4-C2
00567001 5963 F.F. DRIVER TC4-A8
20267179 12AT7 F.F. UNCL. TC4-A3
00667001 5881 F.F. DRIVER TC4-B3
00667001 5881 F.F. DRIVER TC4-C3
20267006 68Q7A PRE-AMP MTC 3/4-A
10667016 5687 DUAL PHASE INVERT. TC1-C1
10667017 68Q7A CLIPPER TC1-D1
20267012 12AT7 F.F. CLAMPED TC1-D2
20267179 12AT7 F.F. UNCL. TC1-B2
00367001 5687 F.F. DRIVER TC1-C2
20267006 68Q7A PRE-AMP MTC 1/2-A
10667016 5687 DUAL PHASE INVERT. TC1-C1
10667017 68Q7A CLIPPER TC1-A2
20267012 12AT7 F.F. UNCL. TC1-A3
00567001 5963 F.F. DRIVER TC1-B3
20267012 12AT7 F.F. CLAMPED TC1-A4
20267012 12AT7 F.F. CLAMPED TC1-B4
00467001 5963 LOG. DRIVER TC1-C5
20267179 12AT7 F.F. UNCL. TC1-D4
00367001 F.F. DRIVER TC1-D5
20267006 68Q7A PRE-AMP MTC 1/3-A
10667016 5687 DUAL PHASE INVERT. TC1-C6
1066017 68Q7A CLIPPER TC1-D6
20267012 12AT7 F.F. CLAMPED TC1-D7
20267179 12AT7 F.F. UNCL. TC1-C7
00367001 5687 F.F. DRIVER TC1-B7
20267006 68Q7A PRE-AMP MTC 1/2-B
10667016 5687 DUAL PHASE INVERT. TC1-C6
10667017 68Q7A CLIPPER TC1-A8
20267012 12AT7 F.F. CLAMPED TC1-B8
20267179 12AT7 F.F. UNCL. TC1-C8
00367001 5687 F.F. DRIVER TC1-D8
20267010 DUAL 12AT7 C. FOLLOWER TC1-A5
00467001 5963 LOG. DRIVER TC1-C3
20267002 12BH7 F.F. CLAMPED MTC 3/4-F2
20267004 5687 RECORD MTC 1/3-E2
F RECORD (WRITE) HEAD
INPUTS FROM F_R RECORD PROPOSITIONS (SEE FIGURE)
20267010 DUAL 12AT7 C. FOLLOWER TC1-A6
00467001 5963 LOG. DRIVER TC1-A7
20267002 12BH7 F.F. CLAMPED MTC-6A 1/3-E1
20267004 5687 RECORD MTC 1/3-E2
H RECORD (WRITE) HEAD
INPUTS FROM H_R RECORD PROPOSITIONS (SEE FIGURE)
20267010 DUAL 12AT7 C. FOLLOWER TC1-B10
00467001 5963 LOG. DRIVER TC1-A7
20267002 12BH7 F.F. CLAMPED MTC 2/4-E1
20267004 5687 RECORD MTC 1/2-F2
J RECORD (WRITE) HEAD
INPUTS FROM J_R RECORD PROPOSITIONS (SEE FIGURE)
E WRITE CIRCUIT
20267010 DUAL 12AT7 C. FOLLOWER TC2-B2
20267010 DUAL 12AT7 C. FOLLOWER TC2-C5
20267010 DUAL 12AT7 C. FOLLOWER TC2-A5
20267010 DUAL 12AT7 C. FOLLOWER TC2-B5
INPUTS FROM E_R RECORD PROPOSITIONS (SEE FIGURE)
00467001 5963 LOG. DRIVER TC4-D3
20267002 12BH7 F.F. CLAMPED MTC 1/2-F1
20267004 5687 RECORD MTC 1/2-F2
E RECORD (WRITE) HEAD
G WRITE CIRCUIT
20267010 DUAL 12AT7 C. FOLLOWER TC1-A5
20267010 DUAL 12AT7 C. FOLLOWER TC1-A6
00467001 5963 LOG. DRIVER TC1-C3
20267002 12BH7 F.F. CLAMPED MTC-2GD 3/4 F1
20267004 5687 RECORD MTC 2/4-E2
G RECORD (WRITE) HEAD
INPUTS FROM G RECORD PROPOSITIONS (SEE FIGURE)
LOGICAL
NOT USED
EFGH

Figure 15. Recirculating Registers

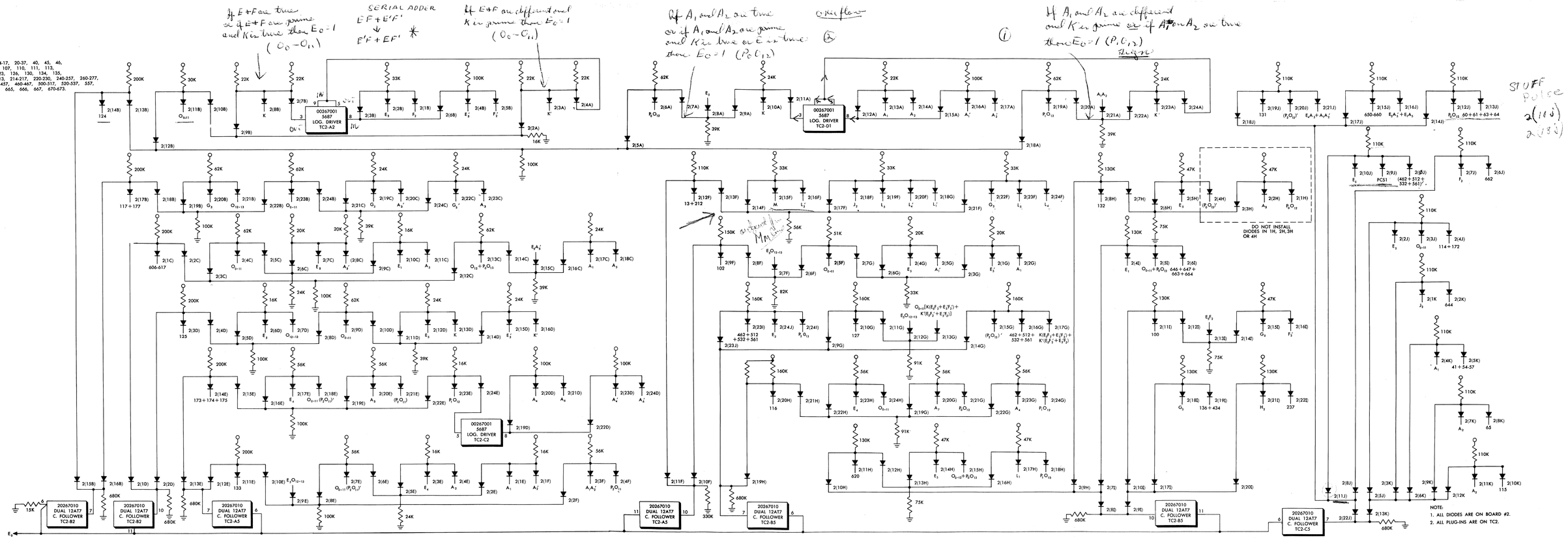
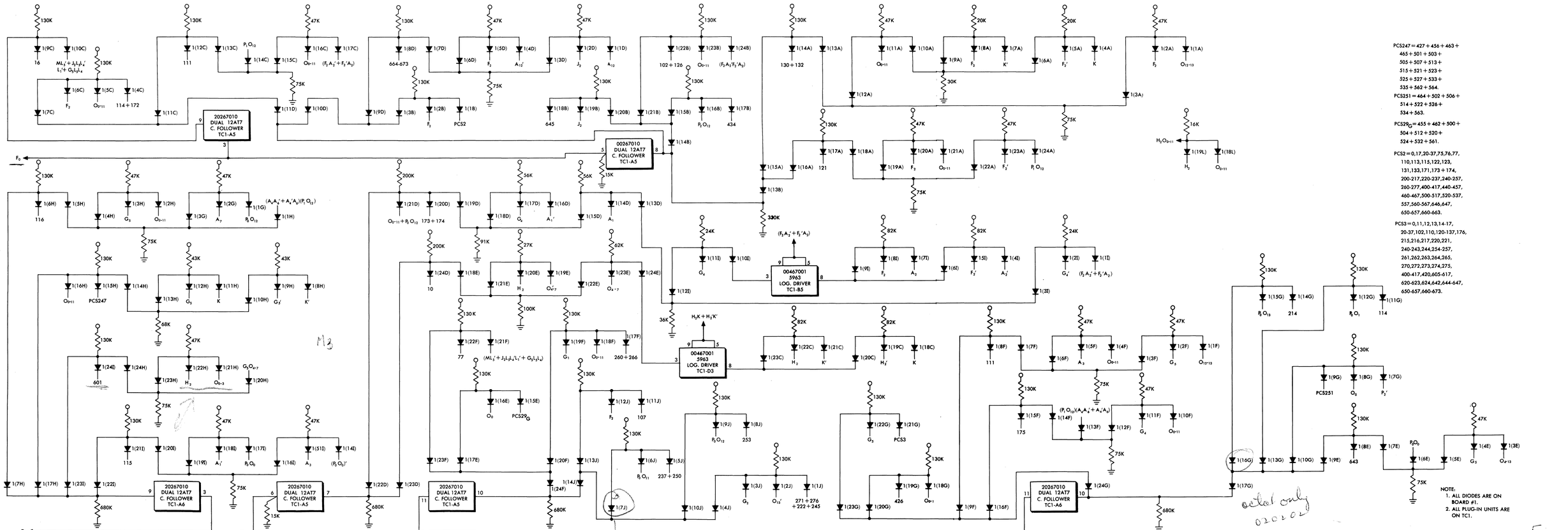

PC51 = 0-7, 14-17, 20-37, 40, 45, 46, 75, 76, 77, 107, 110, 111, 113, 121, 122, 123, 126, 130, 134, 135, 171, 176, 213, 214-217, 220-230, 240-257, 260-277, 400-417, 440-457, 460-467, 500-517, 520-537, 557, 560-567, 645, 665, 666, 667, 670-673.
If E+F are true or if E+F are prime and K is true then $E_O = 1$ $(O_0 - O_{11})$
SERIAL ADDER
EF + E'F'
↓
E'F + EF'
If E+F are different and K is prime then $E_O = 1$ $(O_0 - O_{12})$
If A_1 and A_2 are true or if A_1 and A_2 are prime and K is true or E is true then $E_O = 1$ $(P_0 O_{12})$
overflow
If A_1 and A_2 are different and K is prime or if A_1 and A_2 are true then $E_O = 1$ $(P_1 O_{12})$
00267001 5687 LOG. DRIVER TC2-A2
00267001 5687 LOG. DRIVER TC2-D1
00267001 5687 LOG. DRIVER TC2-C2
20267010 DUAL 12AT7 C. FOLLOWER TC2-B2
20267010 DUAL 12AT7 C. FOLLOWER TC2-A5
20267010 DUAL 12AT7 C. FOLLOWER TC2-B5
20267010 DUAL 12AT7 C. FOLLOWER TC2-C5
DO NOT INSTALL DIODES IN 1H, 2H, 3H OR 4H
STUFF Pulse 2(18J) 2(13J)
NOTE:
1. ALL DIODES ARE ON BOARD #2.
2. ALL PLUG-INS ARE ON TC2.

Figure 16. E_O Logical Network


20267010 DUAL 12AT7 C. FOLLOWER TC1-A5
00267010 DUAL 12AT7 C. FOLLOWER TC1-A5
00467001 5963 LOG. DRIVER TC1-B5
00467001 5963 LOG. DRIVER TC1-D3
20267010 DUAL 12AT7 C. FOLLOWER TC1-A6
20267010 DUAL 12AT7 C. FOLLOWER TC1-A5
20267010 DUAL 12AT7 C. FOLLOWER TC1-A5
20267010 DUAL 12AT7 C. FOLLOWER TC1-A6
octal with addresses 020202
octal only 020202
NOTE:
1. ALL DIODES ARE ON BOARD #1.
2. ALL PLUG-IN UNITS ARE ON TC1.

PCS247 = 427 + 456 + 463 + 465 + 501 + 503 + 505 + 507 + 513 + 515 + 521 + 523 + 525 + 527 + 533 + 535 + 562 + 564.

PCS253 = 464 + 502 + 506 + 514 + 522 + 526 + 534 + 563.

$PCS29_{10}$ = 455 + 462 + 500 + 504 + 512 + 520 + 524 + 532 + 561.

PCS2 = 0,17,20-37,75,76,77, 110,113,115,122,123, 131,133,171,173 + 174, 200-217,220-237,240-257, 260-277,400-417,440-457, 460-467,500-517,520-537, 557,560-567,646,647, 650-657,660-663.

PCS3 = 0,11,12,13,14-17, 20-37,102,110,120-137,176, 215,216,217,220,221, 240-243,244,254-257, 261,262,263,264,265, 270,272,273,274,275, 400-417,420,605-617, 620-623,624,642,644-647, 650-657,660-673.

Figure 17. F_0 and G_0 Logical Networks

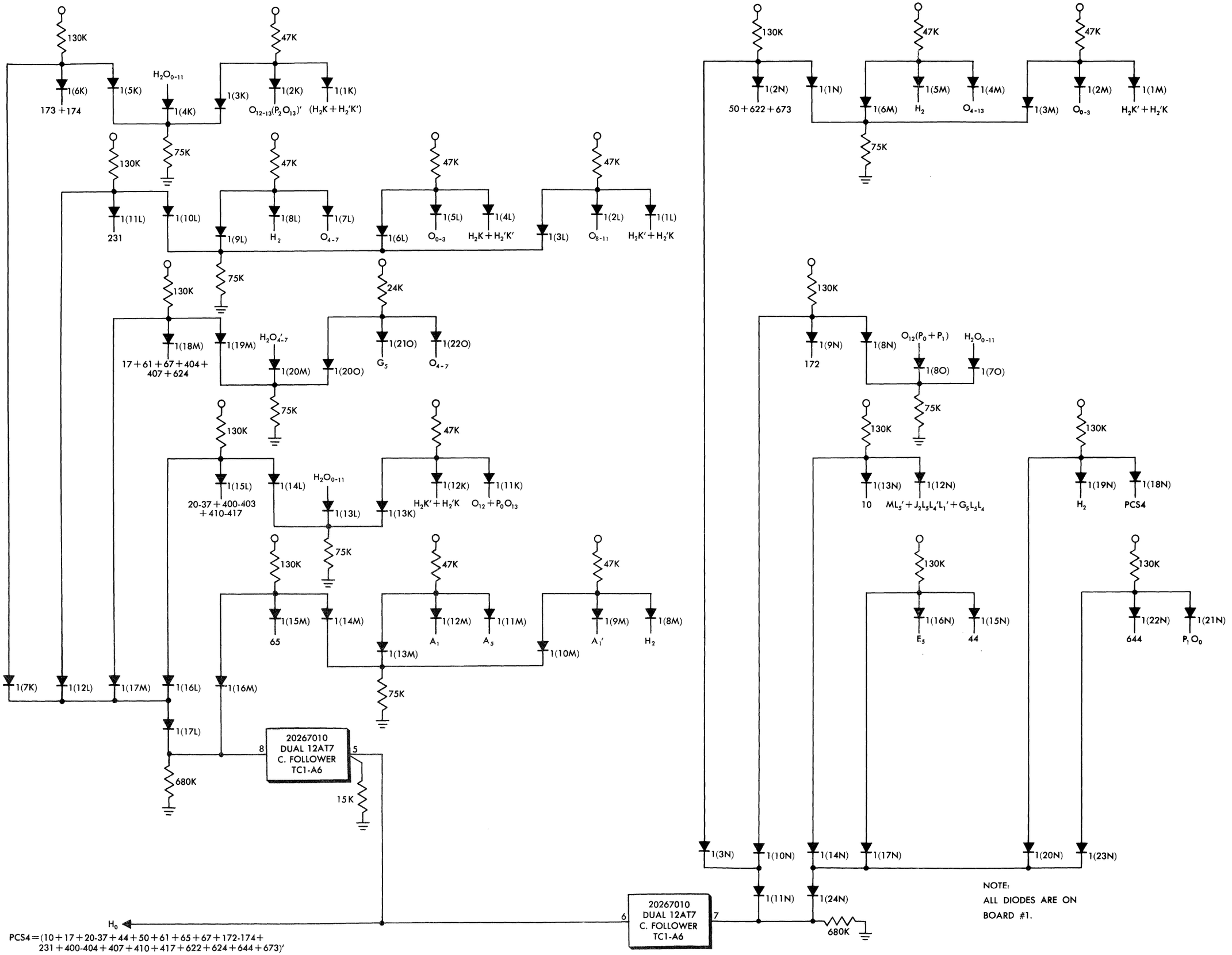


Figure 18. H_0 Logical Network

H_0

(P239)

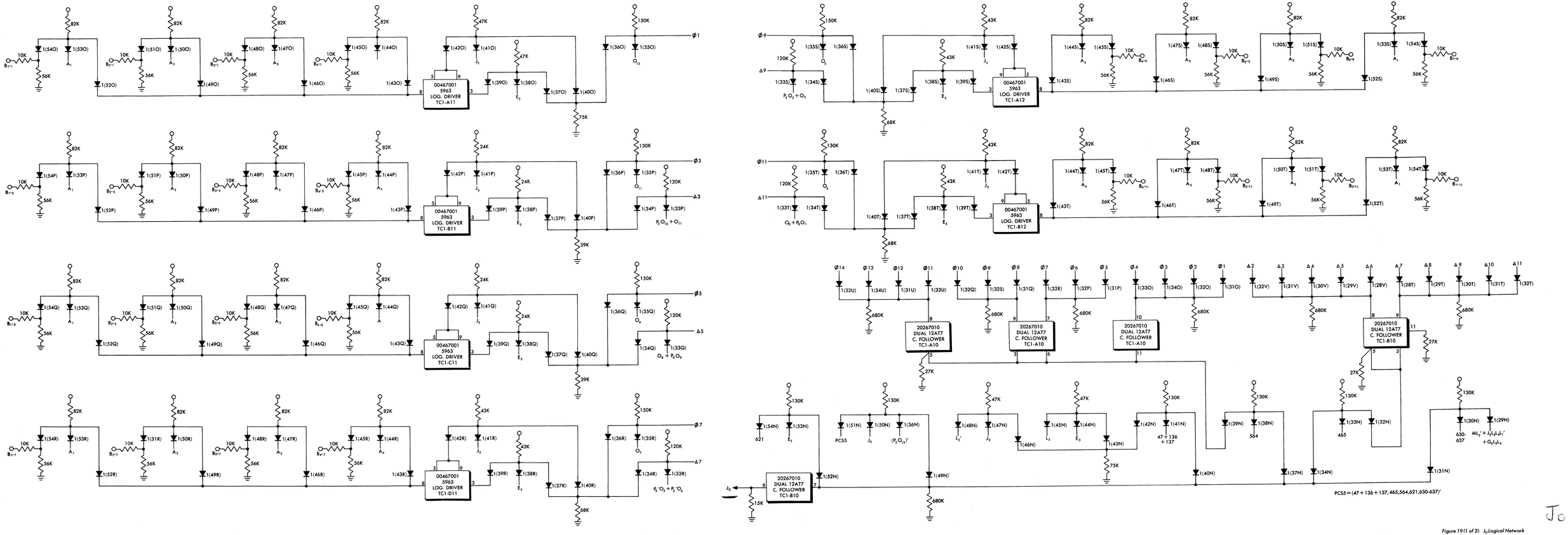
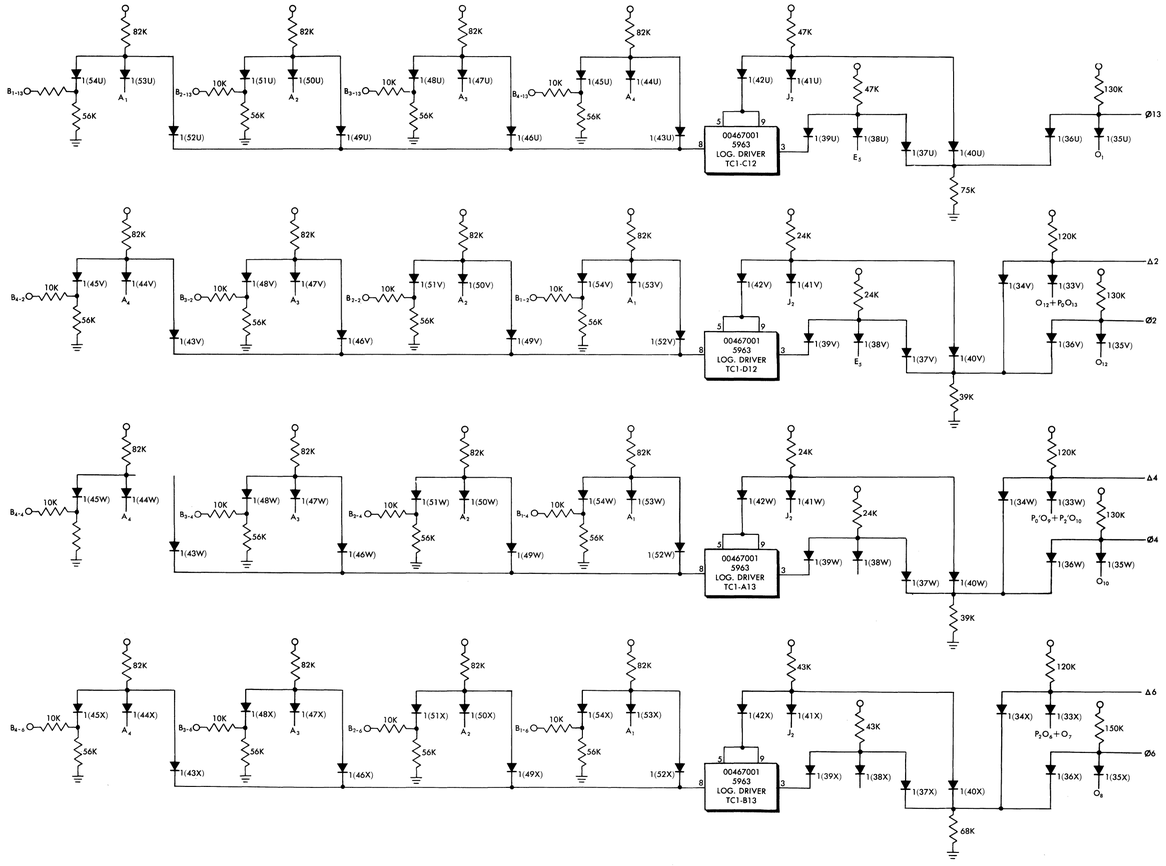


Figure 19(1 of 2). J_0 Logical Network



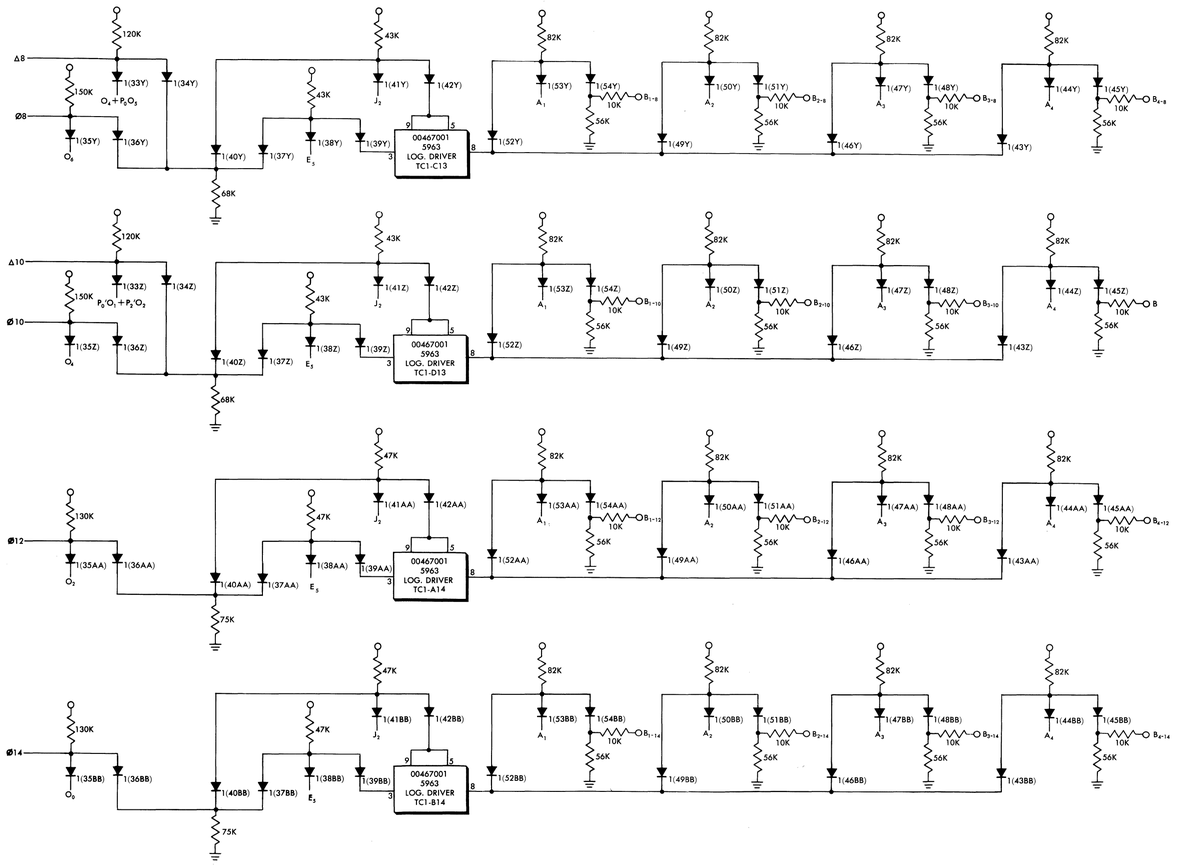


Figure 19 (2 of 2). J_0 Logical Network

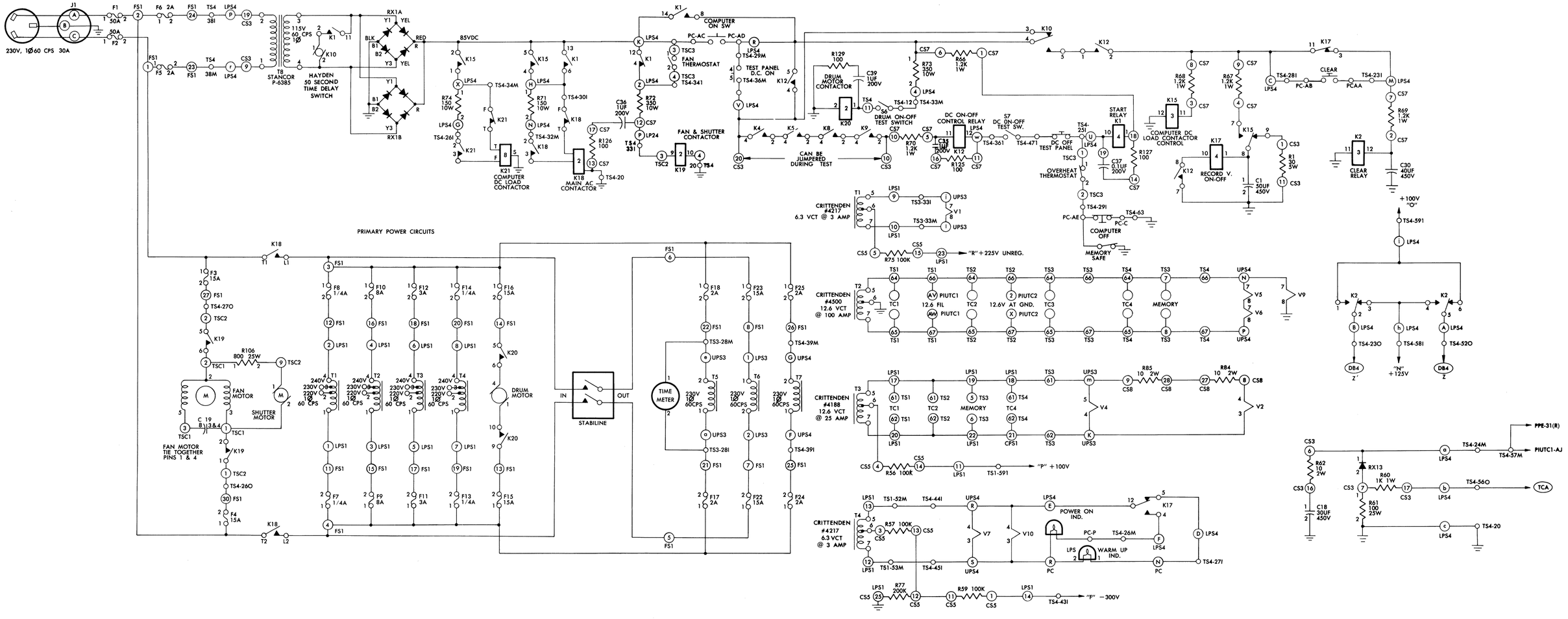


Figure 2011 of 3). Power Supply Schematic

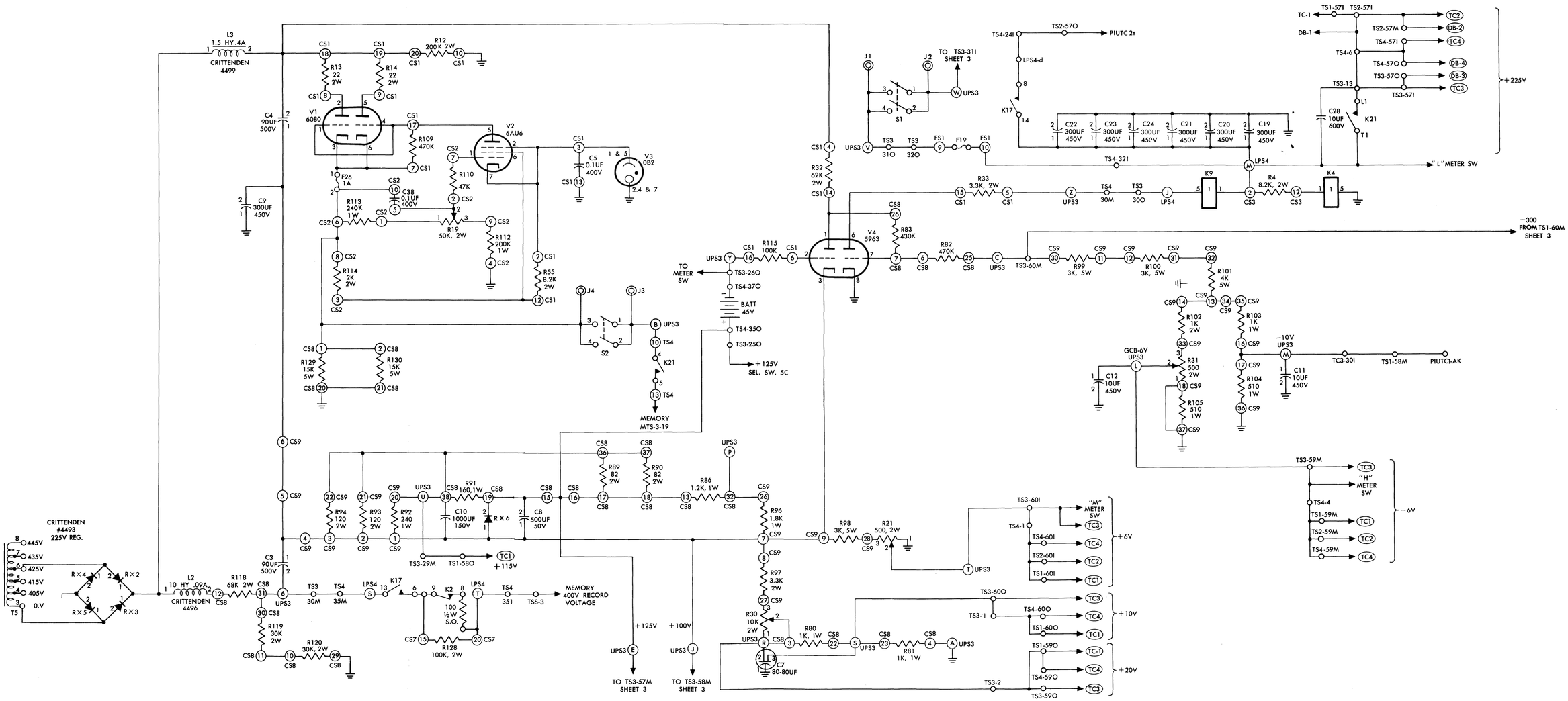


Figure 20 (2 of 3). Power Supply Schematic

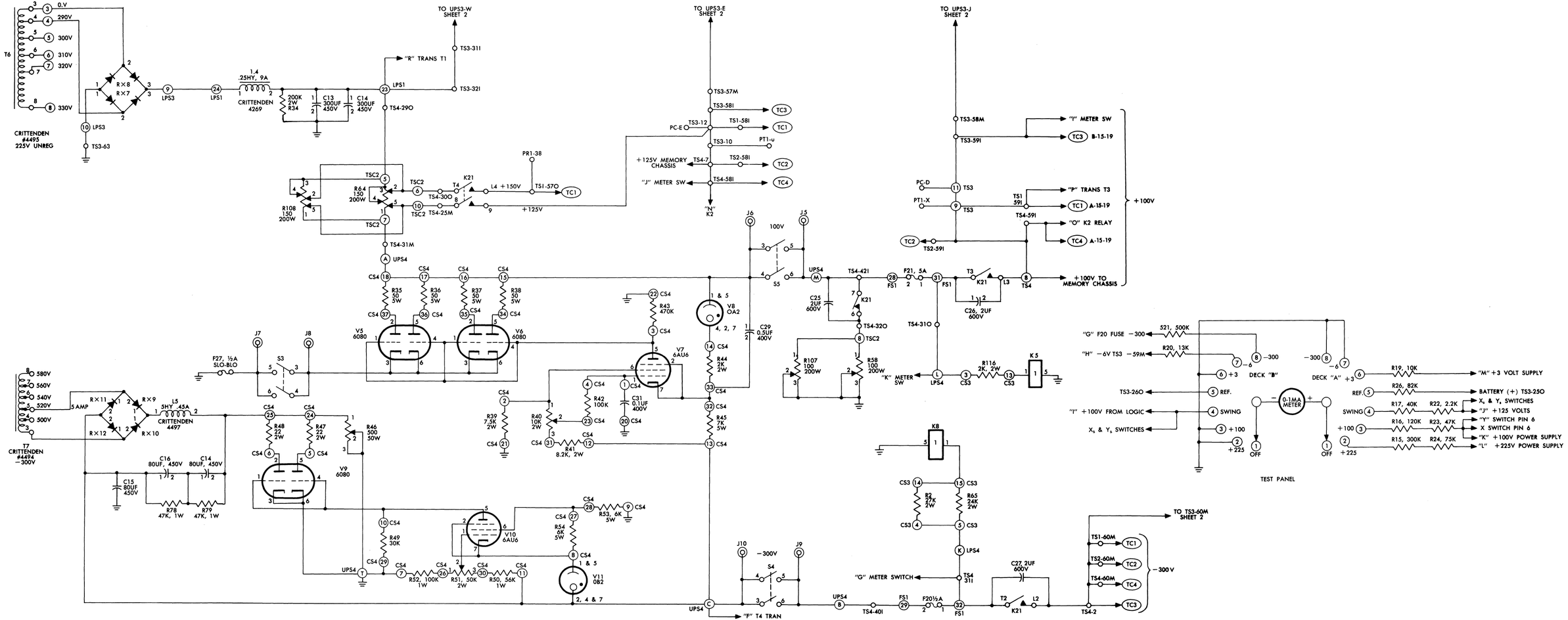
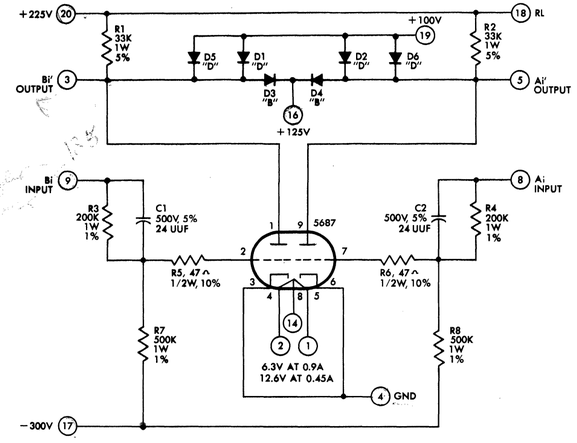
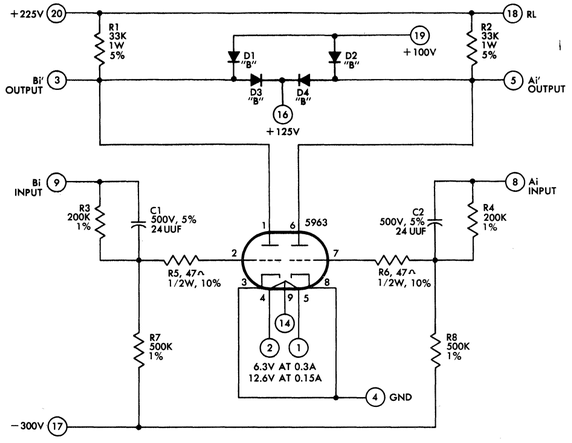


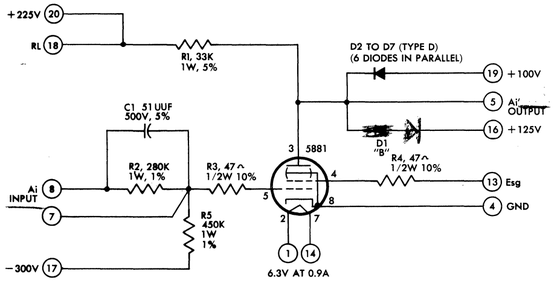
Figure 20 (3 of 3). Power Supply Schematic



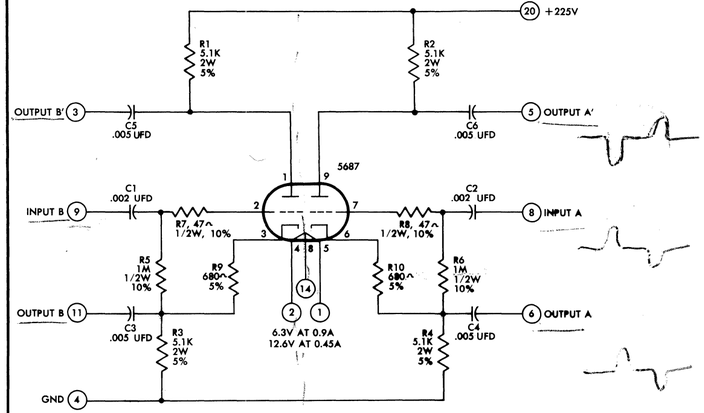
00267001
5687 DRIVER, LOGICAL



00467001
5963 DRIVER, LOGICAL

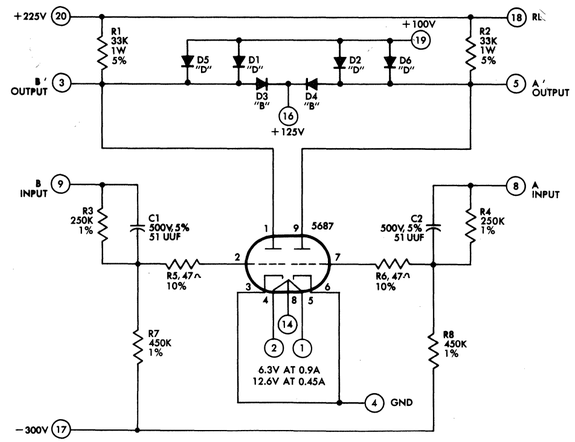


00667001
5881 DRIVER, FLIP-FLOP

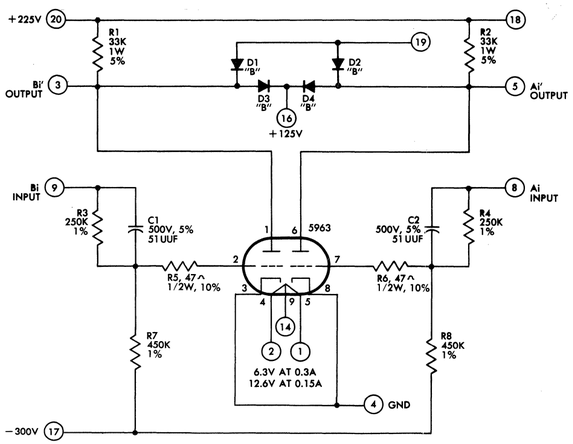


NOTE:
1. R1 & R3 ARE MATCHED TO 1%.
2. R2 & R4 ARE MATCHED TO 1%.

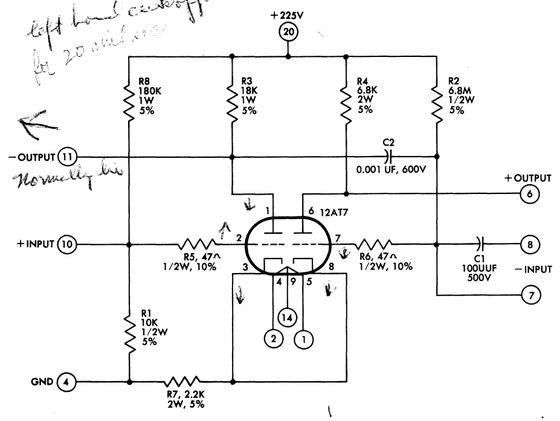
10667016
5687 DUAL PHASE INVERTER



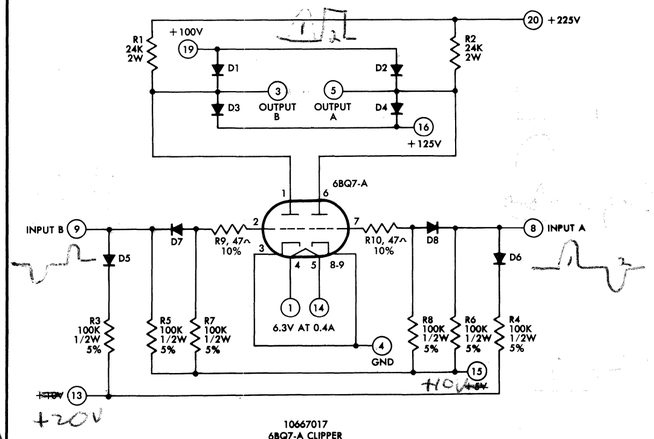
00367001
5687 DRIVER, FLIP-FLOP



00567001
5963 DRIVER, FLIP-FLOP

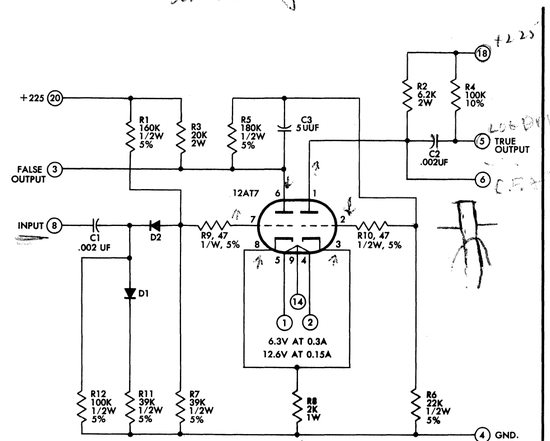


10567002
12AT7 ONE SHOT, UNCLAMPED

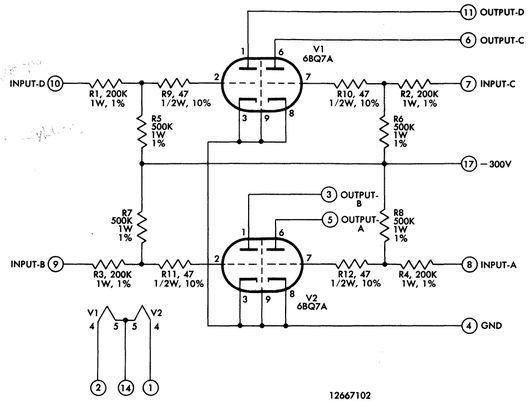


10667017
6BQ7-A CLIPPER

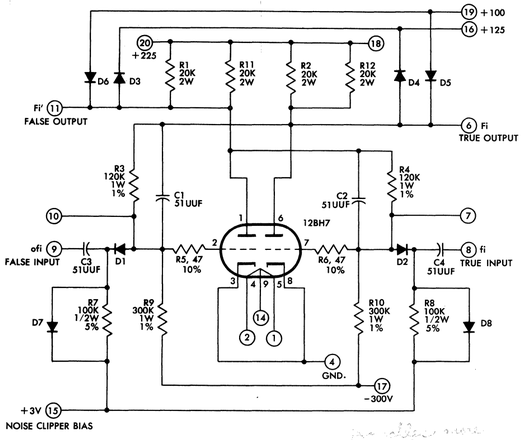
Figure 21 (1 of 3). Plug-in Schematics



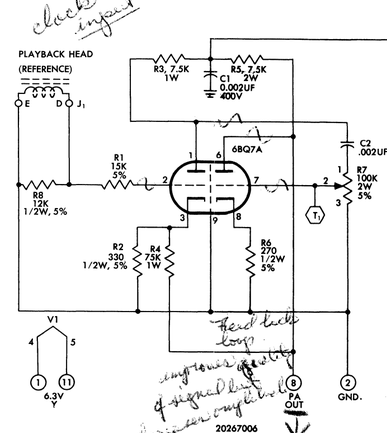
10667018
12AT7 SCHMITT TRIGGER

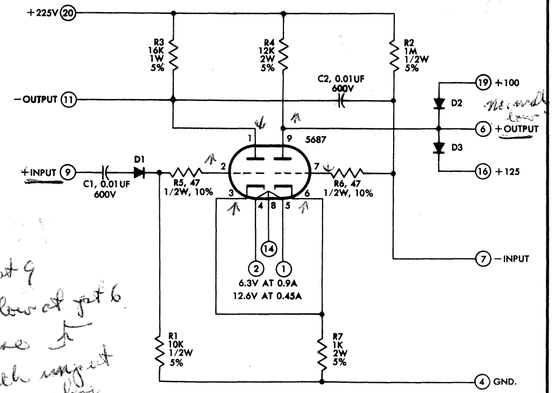


12667102
DUAL 6BQ7A
RELAY DRIVER,
LOGICAL

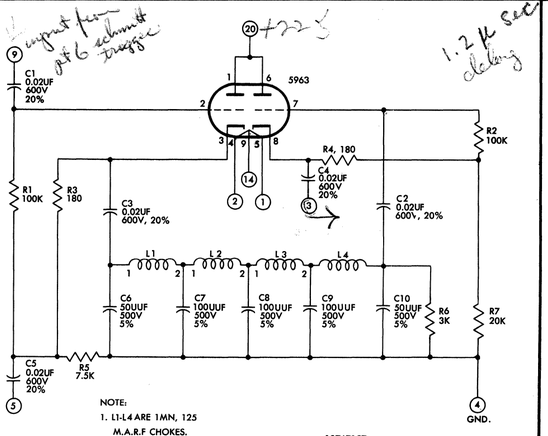


20267002
12BH7 FLIP-FLOP, CLAMPED



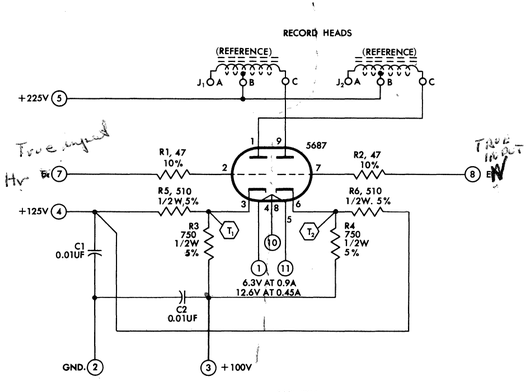


10667029
5687 ONE SHOT, CLAMPED

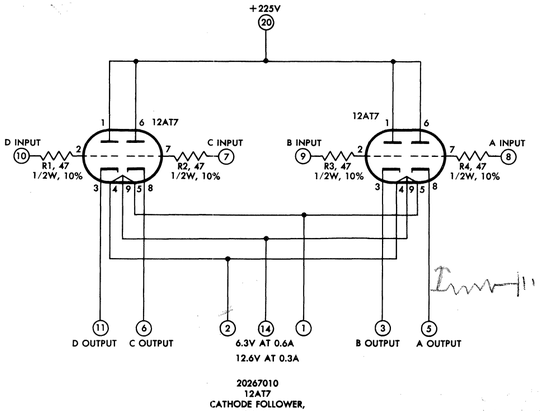


NOTE:
1. L1-L4 ARE 1MN, 125 M.A.R.F CHOKES.
2. ALL RESISTORS 1/2W, 5%.

12767157
DELAY LINE & CATHODE
FOLLOWER

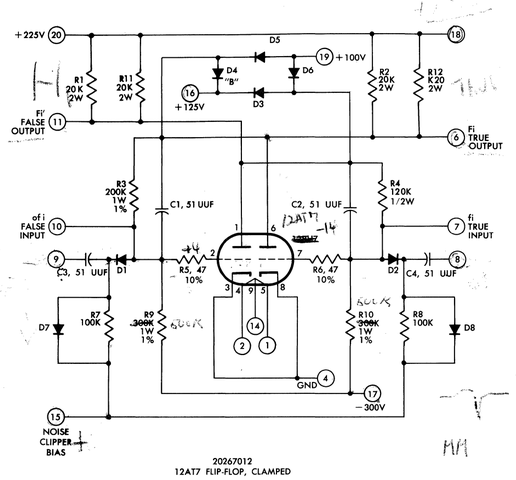


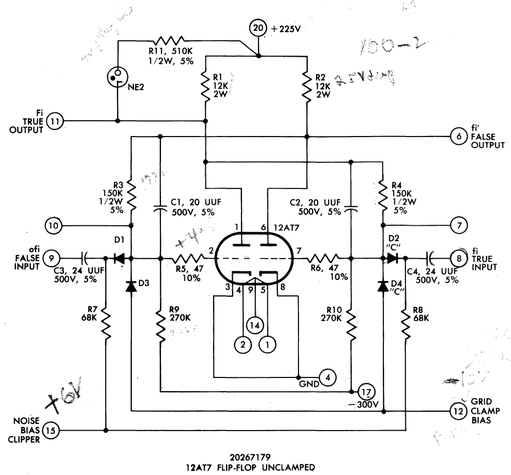
20267004
5687 RECORD

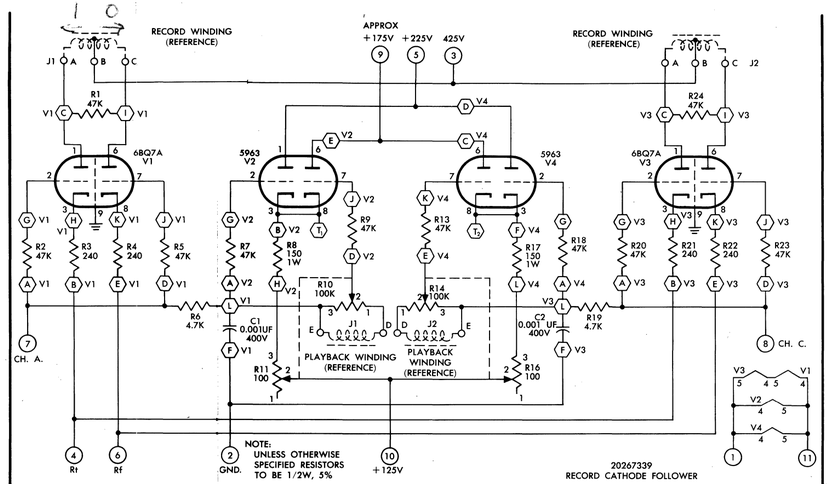


20267010
12AT7
CATHODE FOLLOWER,
LOGICAL

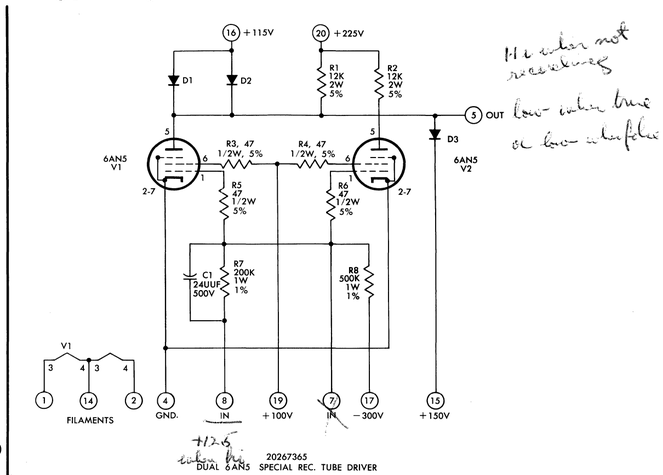
Figure 21 (2 of 3). Plug-in Schematics



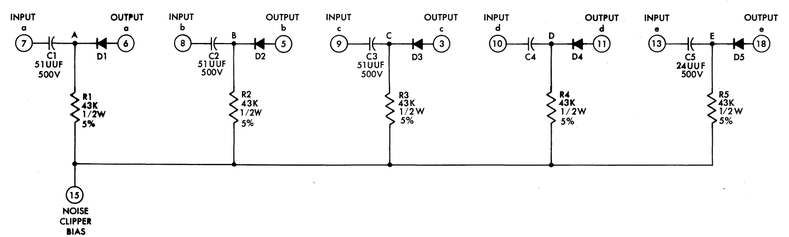




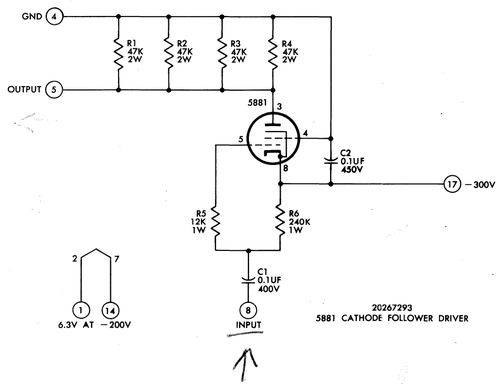
tubes in parallel because of the current 50 ma per tube

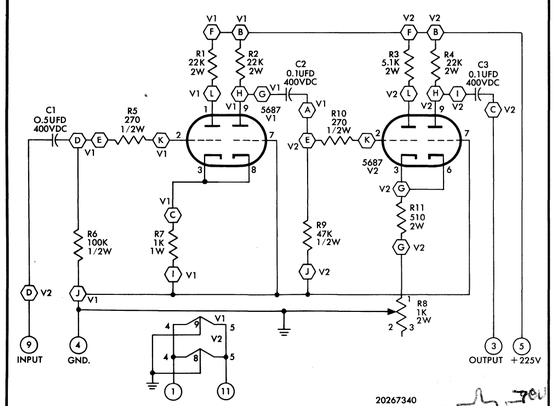


Heater not measured low-value true or two-stage



NOTE:
1. D1-D5 ARE G5519, TYPE C.
2. R1-R5 ARE 43K, 1/2W, 5%.





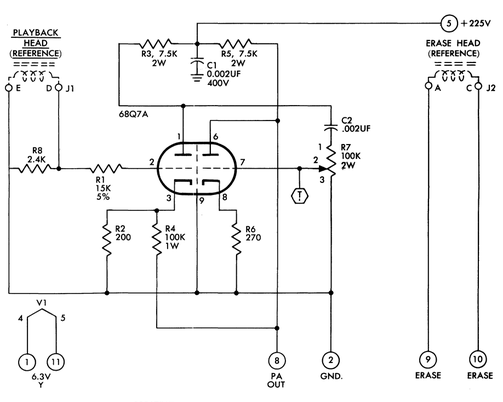
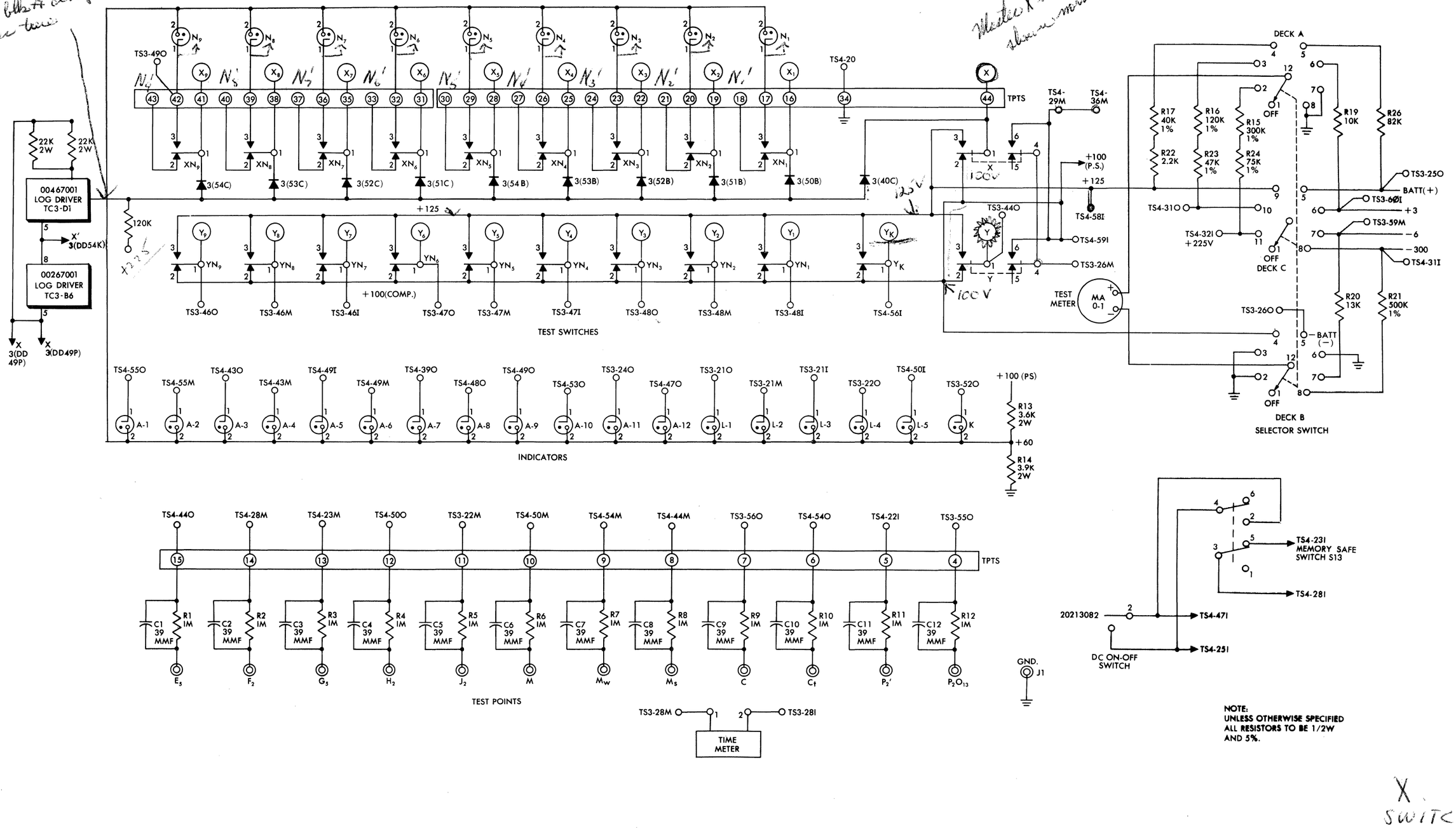


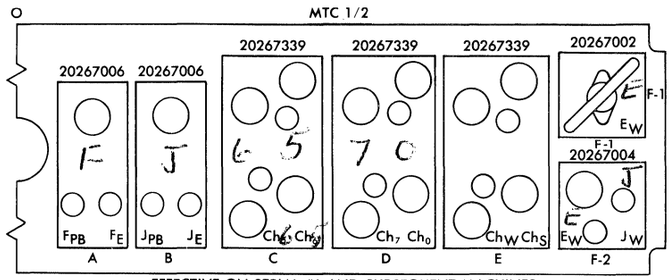
Figure 21 (3 of 3). Plug-in Schematics



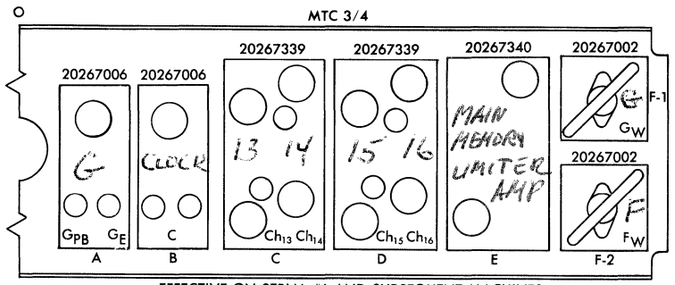
NOTE:
UNLESS OTHERWISE SPECIFIED ALL RESISTORS TO BE 1/2W AND 5%.

Figure 22. Computer Test Panel Schematic

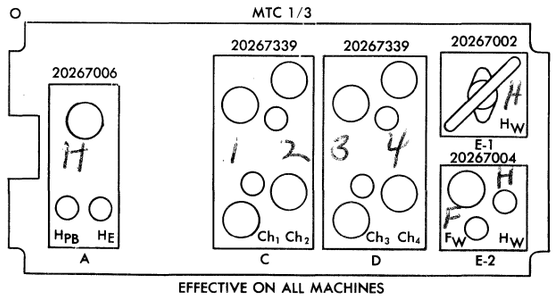
E TC4 A1-8 MTC 2/4 A
F TC1 C1-8 MTC 1/2 A
G TC1 C1-9 MTC 3/4 A
H TC1 C6-8 MTC 1/3 A
J TC1 C6-9 MTC 1/2 B



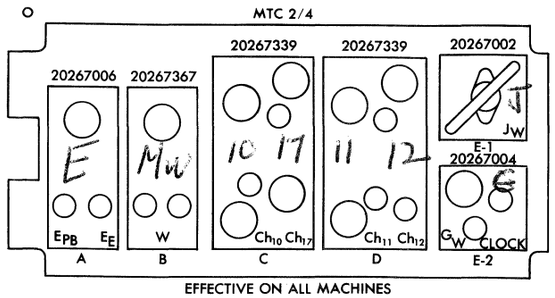
EFFECTIVE ON SERIAL #6 AND SUBSEQUENT MACHINES



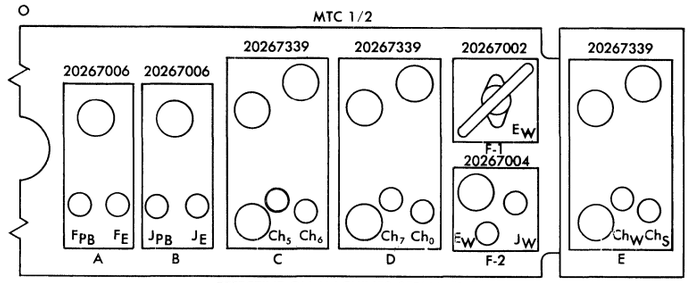
EFFECTIVE ON SERIAL #6 AND SUBSEQUENT MACHINES



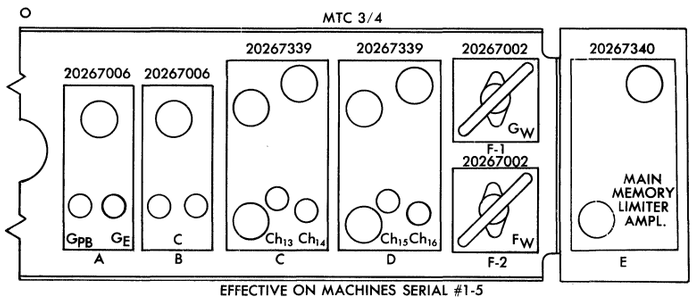
EFFECTIVE ON ALL MACHINES



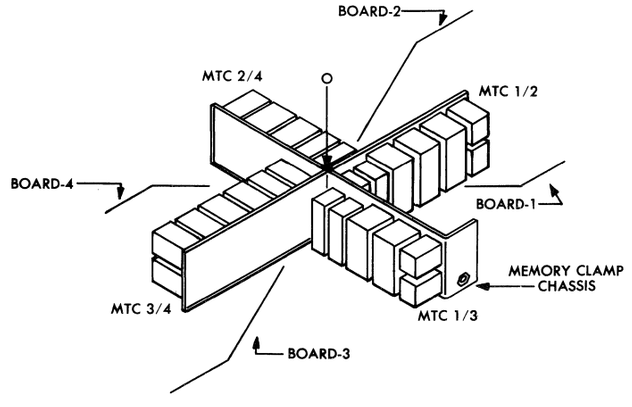
EFFECTIVE ON ALL MACHINES



EFFECTIVE ON MACHINES SERIAL #1-5



EFFECTIVE ON MACHINES SERIAL #1-5



NOTE:
ABOVE FIGURE APPLIES TO SERIAL #6 AND SUBSEQUENT MACHINES

Figure 23. Memory Tube Chassis Component Installation

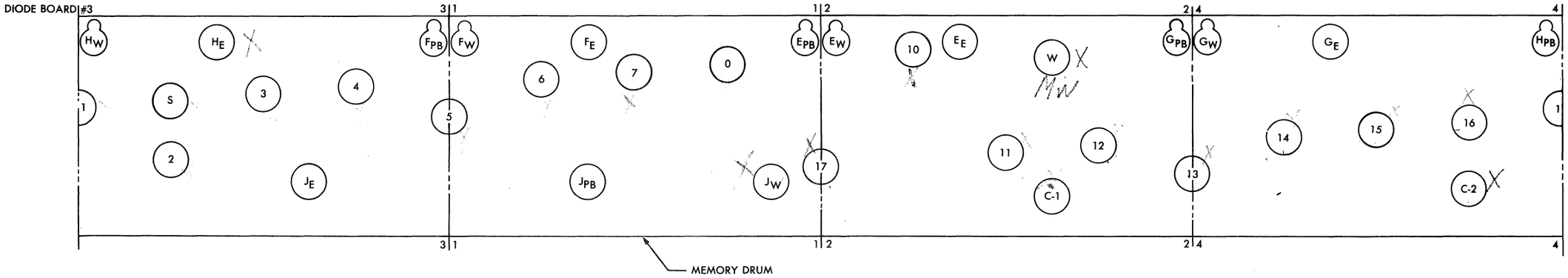


Figure 24. Physical Locations of Read-Write Heads

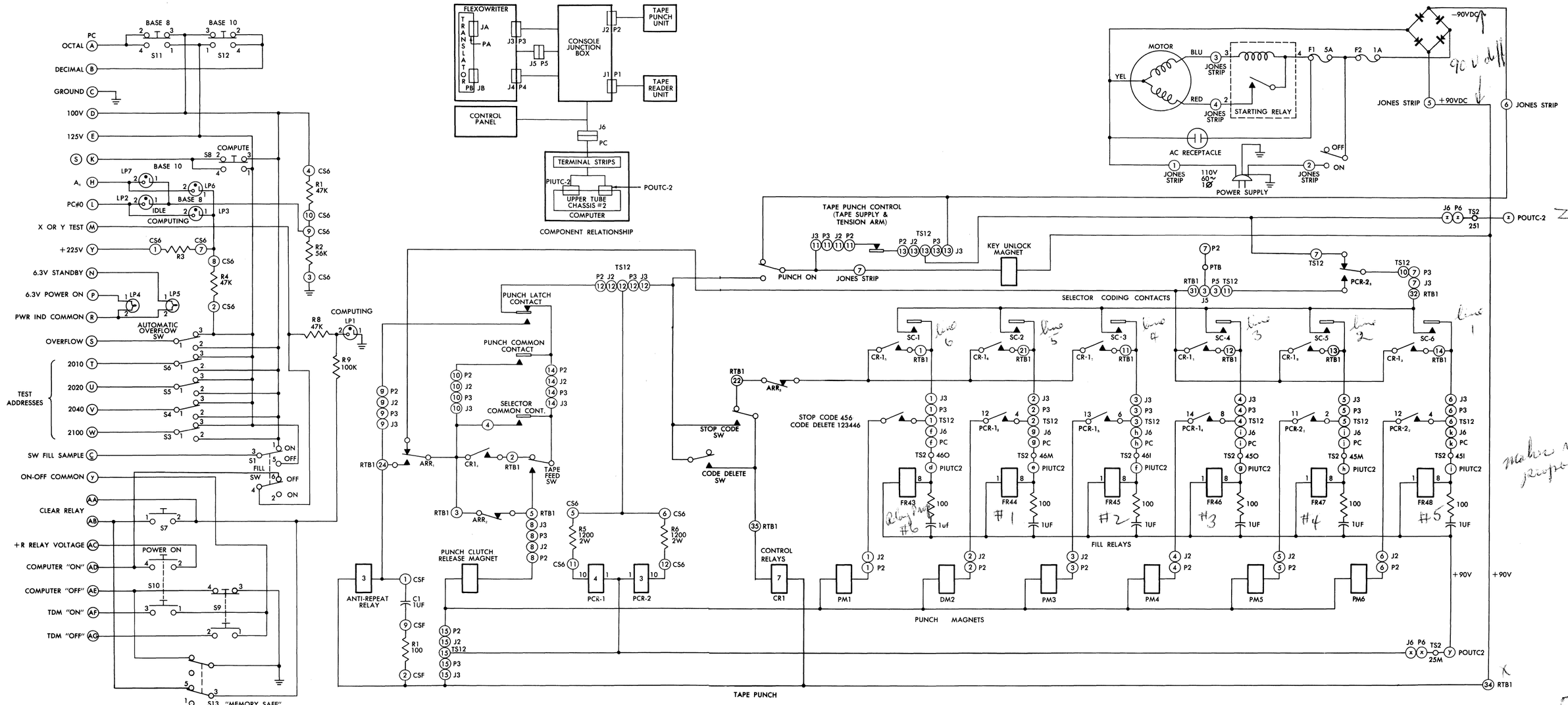


Figure 25(1 of 2). Schematic of Operator's Console

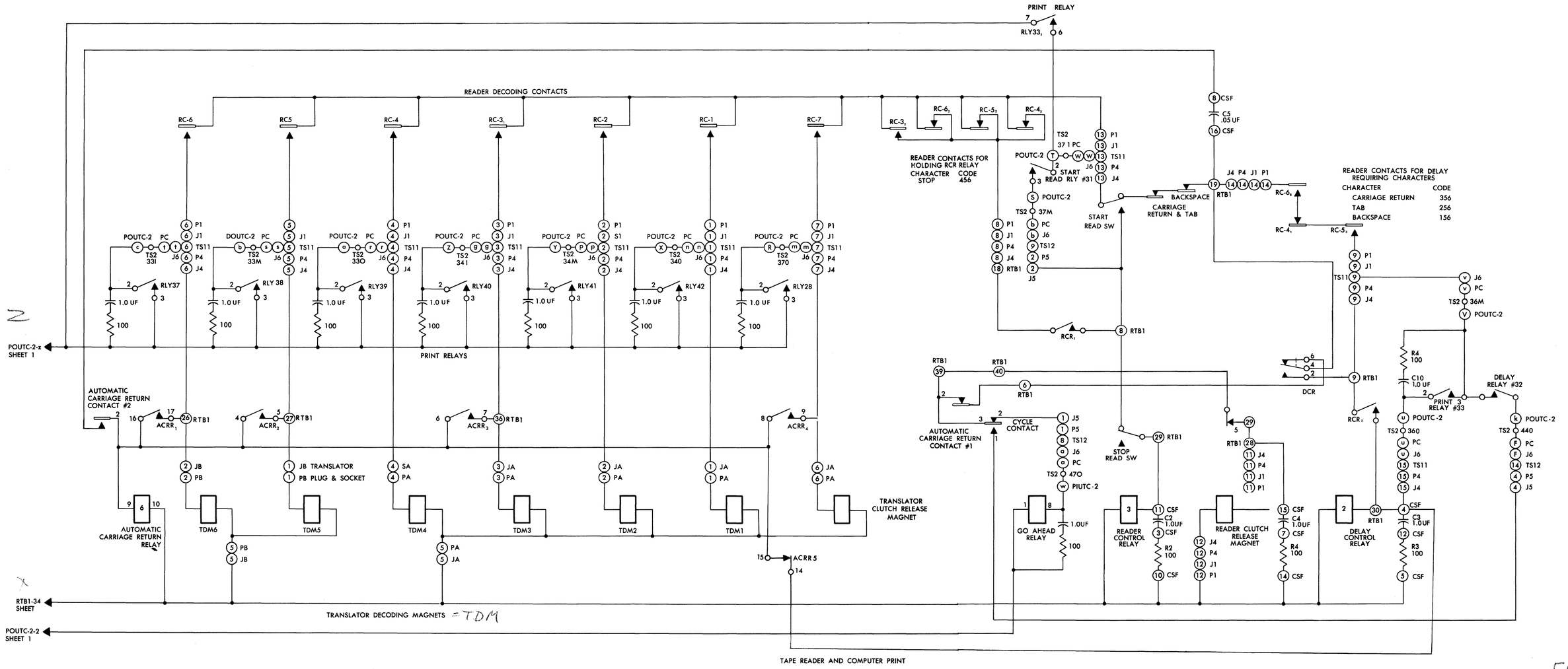
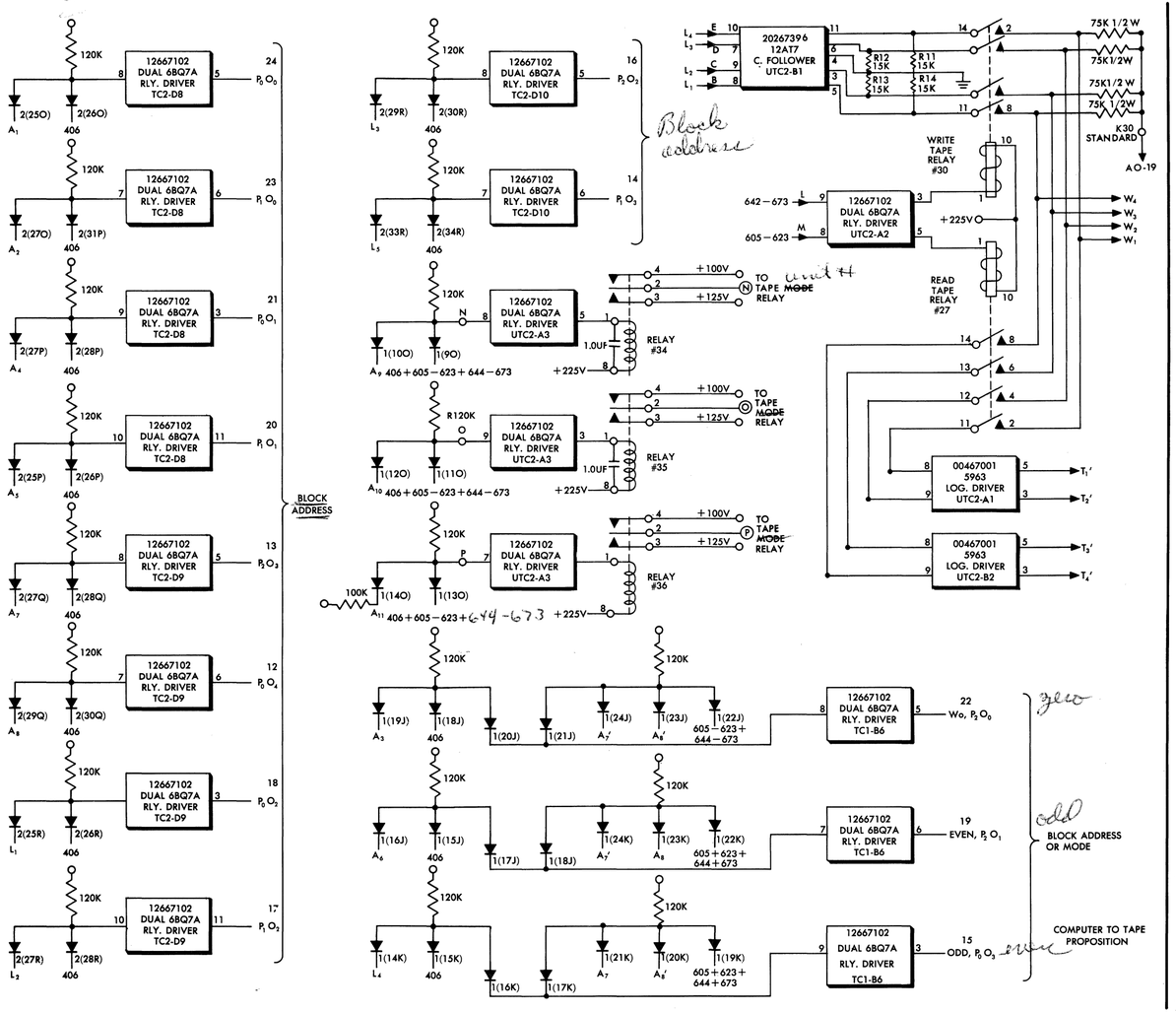
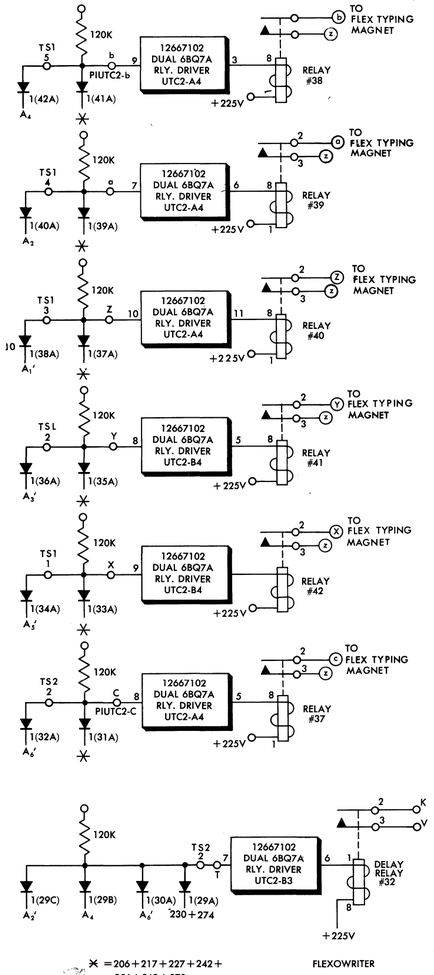
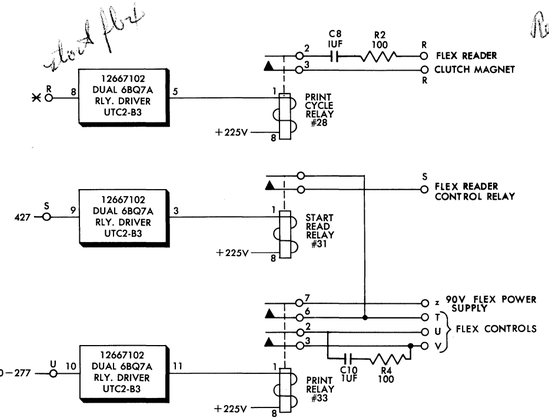
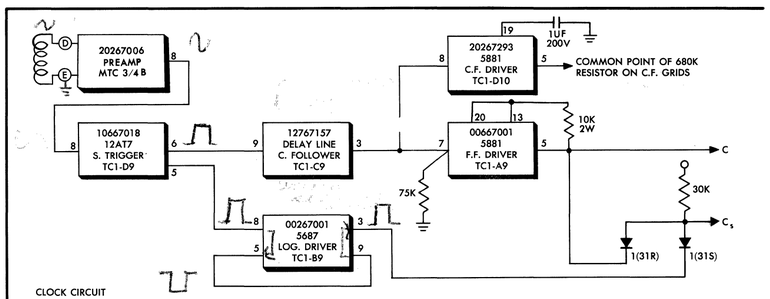


Figure 25(2 of 2). Schematic of Operator's Console









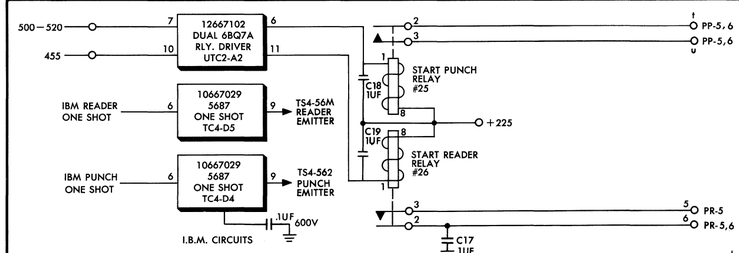
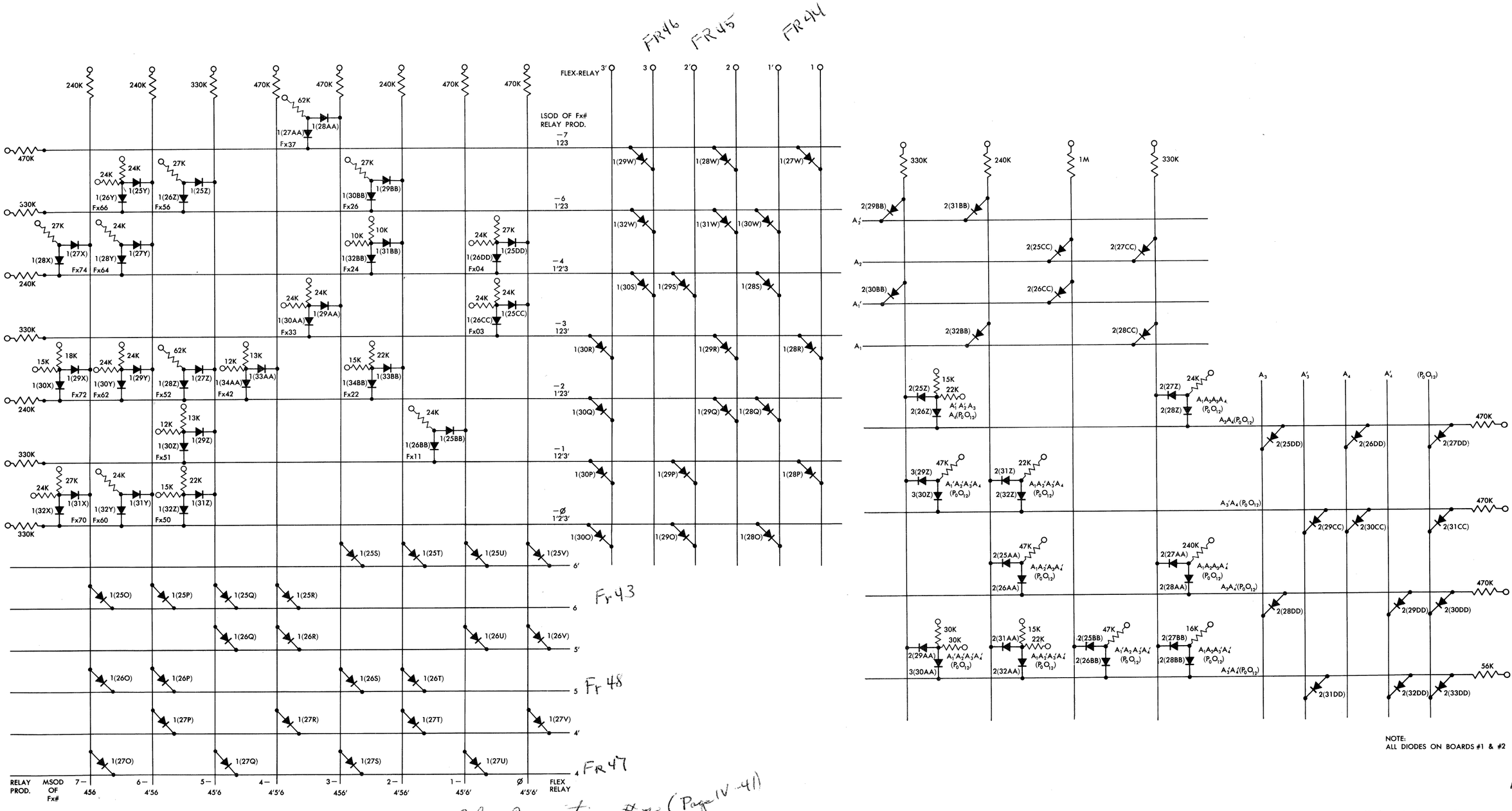
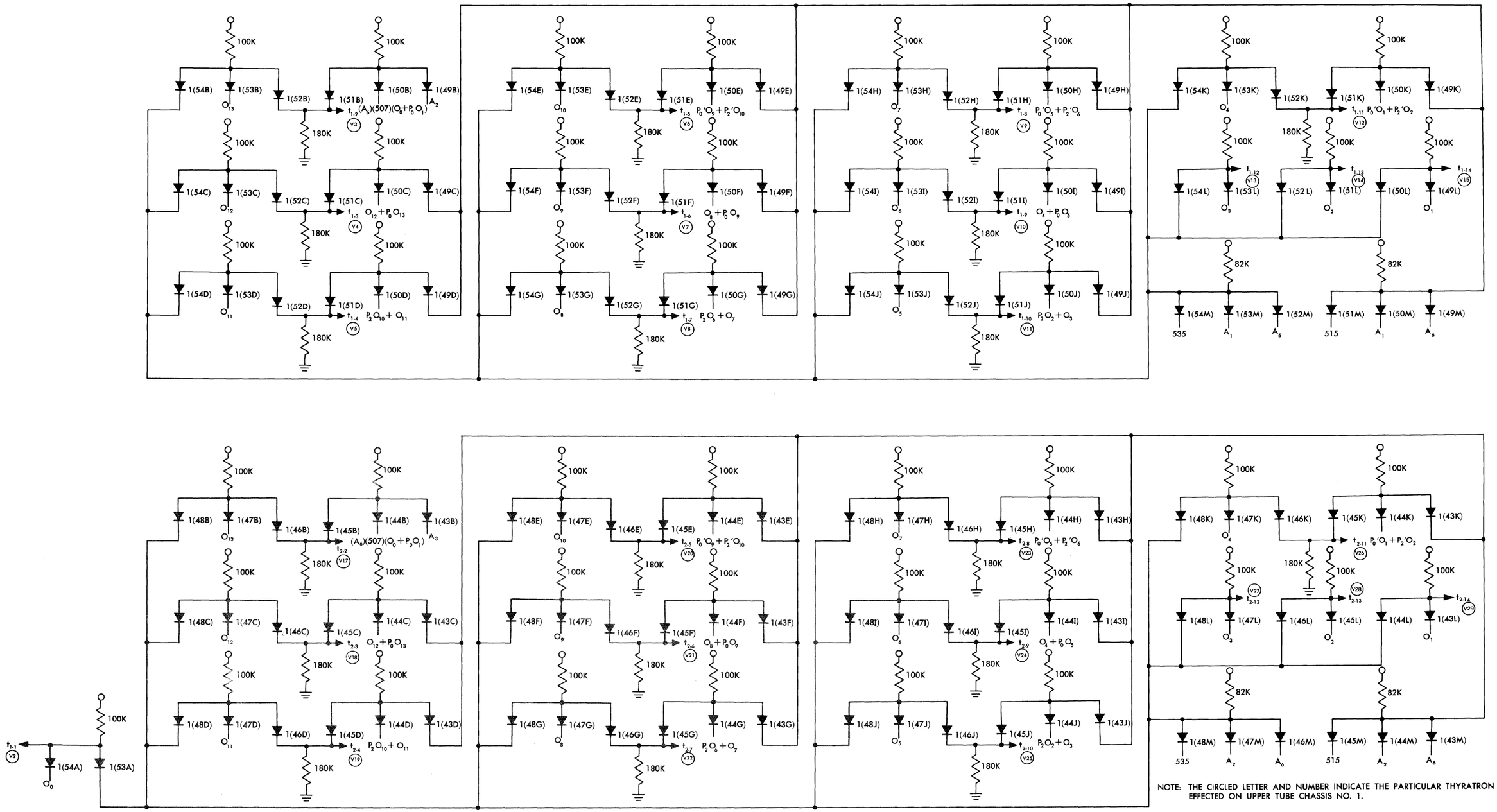


Figure 26. Input-Output Circuitry



Relay Propositions #2 (Page IV-41)

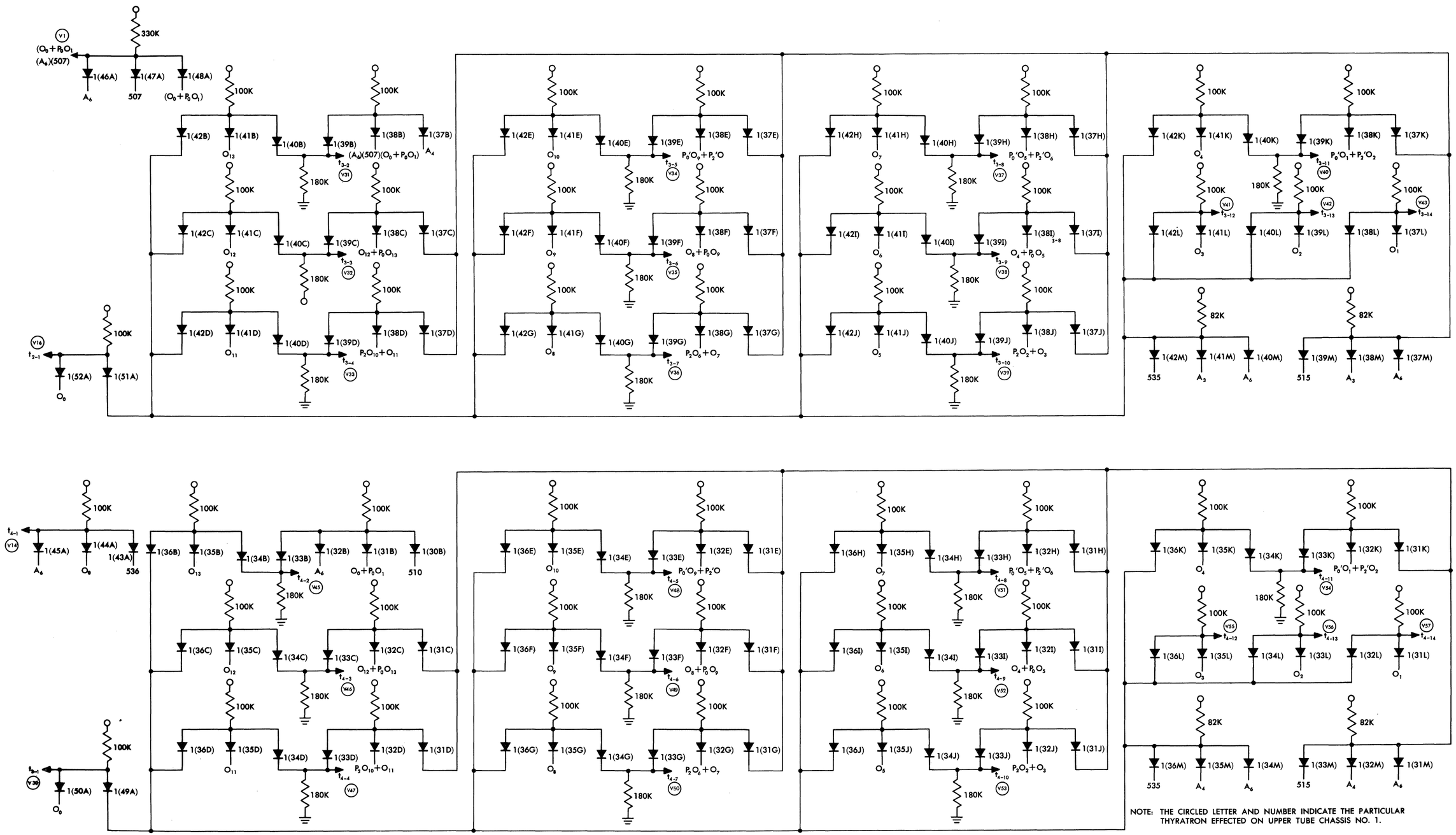
Figure 27. Flexowriter Conversion Circuits



NOTE: THE CIRCLED LETTER AND NUMBER INDICATE THE PARTICULAR THYRATRON EFFECTED ON UPPER TUBE CHASSIS NO. 1.

Figure 28 (1 of 2). Thyratron Grid Logical Networks

IBM



NOTE: THE CIRCLED LETTER AND NUMBER INDICATE THE PARTICULAR THYRATRON EFFECTED ON UPPER TUBE CHASSIS NO. 1.

Figure 28 (2 of 2). Thyratron Grid Logical Networks